# BACHELOR'S THESIS ASSIGNMENT

## I. Personal and study details

Student's name: **Kashyap Suyash** Personal ID number: **471834**

Faculty / Institute: **Faculty of Mechanical Engineering**

Department / Institute: **Department of Instrumentation and Control Engineering**

Study program: **Bachelor of Mechanical Engineering**

Branch of study: **Information and Automation Technology**

## II. Bachelor's thesis details

Bachelor's thesis title in English:

**Automated degreasing application for car window body flange**

Bachelor's thesis title in Czech:

**Automatická aplikace odmašťování rámů oken automobilu**

Guidelines:

Design the solution for automated degreasing application of car window body flange prior to window installation. (Industry 4.0 application)

1. Compare at least two differrent ideas for solution and select the feasible one.
2. Design the complete process for selected solution.
3. Create the simulation of the designed process.

Bibliography / sources:

NEČAS, Martin, et al. Automatic Quality Control Workplace Design. In Proceedings of STČ 2019. 2019.

SIMÕES, Ana Correia; SOARES, António Lucas; BARROS, Ana Cristina. Factors influencing the intention of managers to adopt collaborative robots (cobots) in manufacturing organizations. Journal of Engineering and Technology Management, 2020, 57: 101574.

Name and workplace of bachelor's thesis supervisor:

**Ing. Pavel Trnka, Ph.D., U12110.3**

Name and workplace of second bachelor's thesis supervisor or consultant:

Date of bachelor's thesis assignment: **29.10.2021** Deadline for bachelor thesis submission: **20.01.2022**

Assignment valid until: ______________

| Ing. Pavel Trnka, Ph.D.<br>Supervisor's signature | Head of department's signature | prof. Ing. Michael Valášek, DrSc.<br>Dean's signature |
|---|---|---|

## III. Assignment receipt

The student acknowledges that the bachelor's thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the bachelor's thesis, the author must state the names of consultants and include a list of references.

| Date of assignment receipt | Student's signature |
|---|---|

Bachelor Thesis

Czech Technical University in Prague

F2

Faculty of Mechanical Engineering
Department of Instrumentation and Control Engineering

# Automated degreasing application for car window body flange

Suyash Kashyap

Supervisor: Ing. Pavel Trnka, Ph.D.
Field of study: Instrumentation and Automatic Control
January 2022

## Acknowledgements

I would like to express utmost of gratitude towards the supervisor for my thesis, Ing. Pavel Trnka, Ph.D. for his immense patience and support in helping me deliver this thesis. From the inception of the topic to the very delivery of the final work, his guidance was always there when needed.

I would like to also say many thanks to Mr. Jan Smola and Mr. Jaromir Bucanek from TMMCZ Kolin for introducing me to the problem to be solved in this thesis and for their immense support and guidance.

I would also like to thank Mr. Josef Ponikelsky and Mr. Petr Smolik from HennLich s.r.o. for providing me with simulation software and for doing some tests needed during this thesis.

Last but not the least, a huge token of thanks to all my teachers for their contributions towards my educational development and Czech Technical University as a whole for giving me the chance of studying at this university and deliver this work.

I would also like to thank my family members and partner for their unreserved support and motivation.


## Declaration

I declare that I elaborated this thesis on my own and that I mentioned all the information sources that have been used in accordance with the Guideline no. 1/2009 for adhering to ethical principles in the course of elaborating an academic final thesis.

I also declare the use of resources of Toyota Motor Manufacturing Czech Republic s.r.o. to gather information used for this thesis given my current engineering job there.

In Prague . . .

Suyash Kashyap . . .

# Abstract

Industrial automation is crucial for the growth of potential of any industry as well as its employees.In this thesis, the focus is on understanding the benefits and potential of industrial automation technologies and to propose an automated solution for a given problematic process in one specific industry. To achieve this, the author studied the concepts of industrial automation and its technologies, then used them to design and propose an automated solution and then simulated its feasibility in real-time application. The result is the final proposed design of the complete automated system which can be given to any automation technology company to implement in real-life application...

**Keywords:** automation, cobots, linear unit, gripper, design, degreasing, wiping, wipes, industry, simulation, application, technology, process, machine

**Supervisor:** Ing. Pavel Trnka, Ph.D. U12110.3

# Abstrakt

Průmyslová automatizace je klíčová pro růst potenciálu každého odvětví i jeho zaměstnanců. V této práci je kladen důraz na pochopení výhod a potenciálu technologií průmyslové automatizace a na návrh automatizovaného řešení pro daný problematický proces v jednom konkrétním odvětví. K dosažení tohoto cíle autor studoval koncepty průmyslové automatizace a jejích technologií, poté je použil k návrhu a návrhu automatizovaného řešení a následně simuloval jeho proveditelnost v aplikaci v reálném čase. Výsledkem je konečný navržený návrh kompletního automatizovaného systému, který může být poskytnut jakékoli společnosti v oblasti automatizační techniky k implementaci do reálné aplikace...

**Klíčová slova:** automatizace, koboty, lineární jednotka, chapadlo, design, odmašťování, stírání, utěrky, průmysl, simulace, aplikace, technologie, proces, stroj

**Překlad názvu:** Automatická aplikace odmašťování rámů oken automobilu

# Contents

# Figures

# Tables

# Chapter 1

# Introduction

The aim of this thesis is to propose a solution for automating a given process in the Assembly line at **TMMCZ** (**Toyota Motor Manufacturing Czech Republic**) in Kolin, Czech Republic. The given process is the degreasing of the car body flange where windshield glass and back window glass are installed. Description of components of degreasing process at TMMCZ Kolin:-

- **TMMCZ:** this refers to the car manufacturing plant of **Toyota Motors** in the industrial town of Kolin in Czech Republic known as **Toyota Motor Manufacturing Czech Republic** comprising of Body shop, Paint shop, Welding shop and Assembly shop. Two car models - ***Aygo X***(A-segment) and ***Yaris*** (B-segment) are manufactured and assembled in this plant.[46]
- **Degreasing:** this refers to the process of cleaning the car body flange with cleaning agent (liquid IPA (isopropyl alcohol)) and a cleaning cloth/wipe. This is a necessary step to prepare body flange surface for urethane adhesion for the later process of windshield and back window glass installation as the body flange should be free of dirt, dust, grease, oil or fingerprints which may come from previous processes in the Assembly line.[71]
- **Body flange:** this refers to the exact area on the front side (*referred to as FR in TMMCZ*) and back side (*referred to as RR in TMMCZ*) of the car body frame where the windshield and back window glass are installed later (see figure 1.5. This area of the body is already painted (in the Paint shop) before entering into the Assembly line.[71]
- **Liquid IPA:** this refers to the liquid isopropyl alcohol (99 percent IPA concentration) which is used as the cleaning solvent in order to avoid any damage to the body paint while also reacting with the surface conditioners in the body paint to prepare the surface for the urethane adhesion for the installation of the glass in a later process.[71] 99 percent IPA concentration is preferred over 70 percent concentration as it's more effective in removing the really fine impurities from the surface. Many

other automotive manufacturers such as BMW use primer instead of liquid IPA for the degreasing purpose.

- **Cleaning cloth/wipe:** this refers to the cloth piece or industrial wipe (also, known as ***cloth gauze*** in TMMCZ) which is wetted with the liquid IPA to wipe the surface of the body flange in a contact-based cleaning to efficiently remove dirt, dust, grease, oil or fingerprints accumulating on the body flange from the previous processes in the Assembly line.[71]

## 1.1 Current application of degreasing process at TMMCZ

The author describes workplace, manual process and problems for current degreasing:

### 1.1.1 Workplace of degreasing

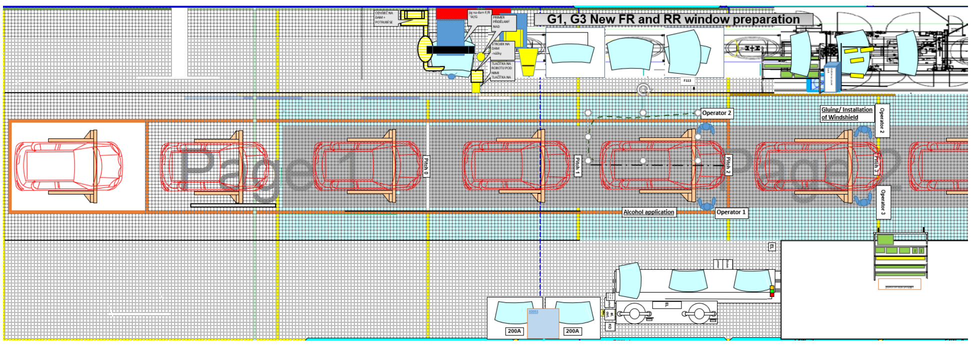


**Figure 1.1:** General layout of the Assembly line for degreasing workplace from top view.[62]

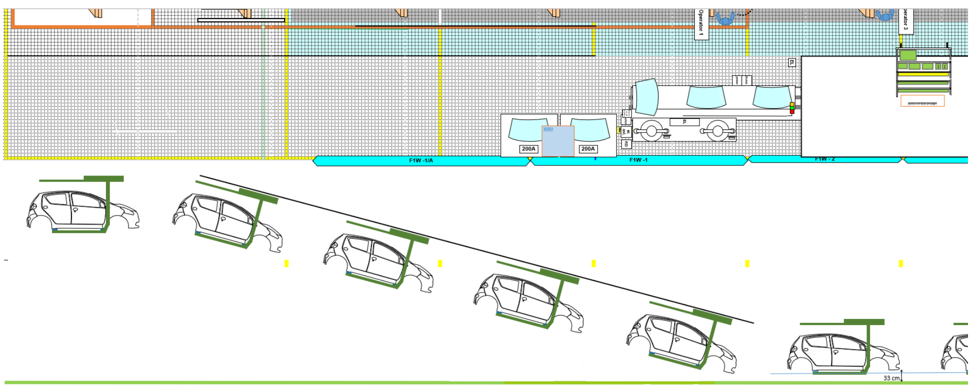

**Figure 1.2:** General layout of the Assembly line for degreasing workplace from side view.[62]

The Assembly line brings car bodies from an upper 'buffer zone' (waiting area) to the lower base floor for degreasing through a tilted path so that the operators can easily stand straight onto the car body-carriers while cleaning the body flange (see figure 1.1). This position is right after the position on the floor where the angled Assembly line becomes parallel to the base floor (see figures 1.2 and 1.4). The angle of tilt of the line from the base floor here at this position is 15 degrees (see figure 1.3).

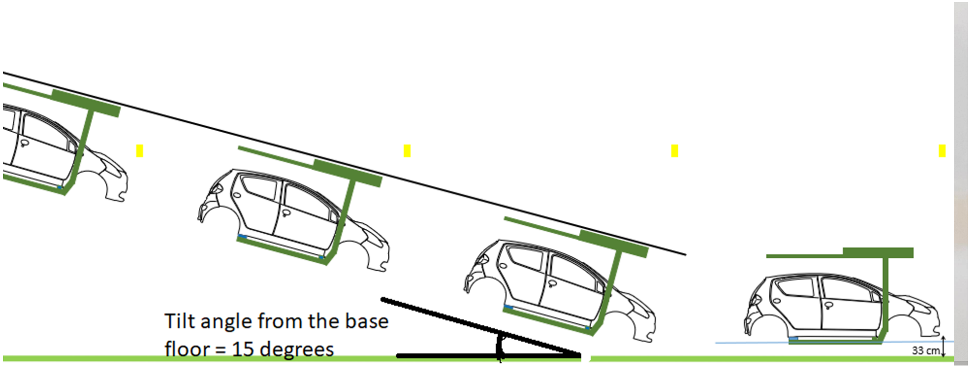


**Figure 1.3:** The tilted Assembly line is at 15 degrees from the base floor.[62]

Just after the **workplace of degreasing of body flange (Pitch 2)**, comes the **workplace of installation of windshield and back window glass (Pitch 3)** (see figure 1.4).

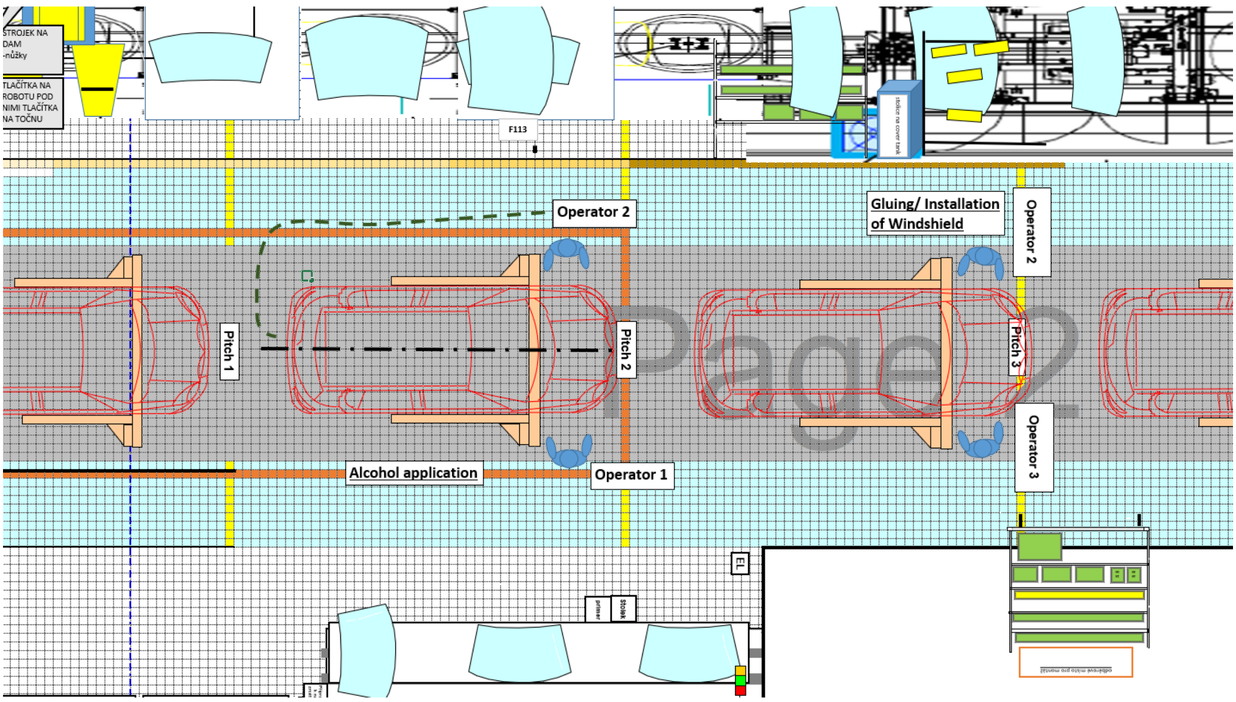


**Figure 1.4:** Workplaces for body flange degreasing and glass installation.[62]

### 1.1.2 Manual degreasing

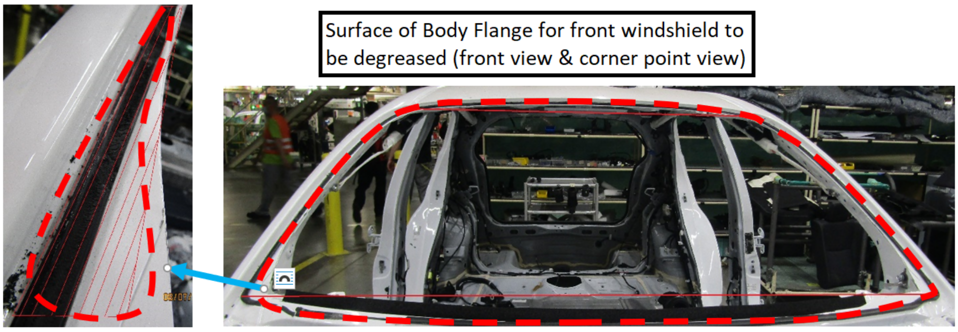


**Figure 1.5:** Body flange for front windshield.[57]

Degreasing of car body flange is currently done by 2 human operators after they climb and stand on the car body carriers' pillars (see the red circled area in figure 1.6). They pick up a cloth piece, wet it with liquid IPA from dispenser and manually wipe the surface of the body flange (see figure 1.5, wiping path marked red) for the full length on which the windshield and back window glass will be installed later (see figure 1.7 for visualisation of the process). This whole degreasing process for front windshield and back window body flange takes 44 seconds in total.[71]

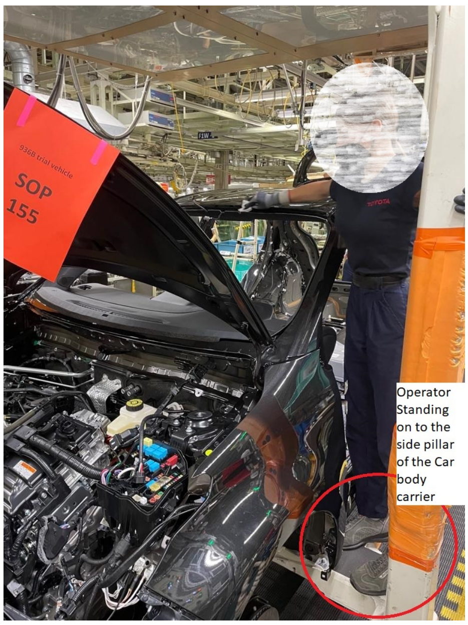


**Figure 1.6:** Operator standing on the side pillar of car body carrier ready to wipe the body flange surface.[57]

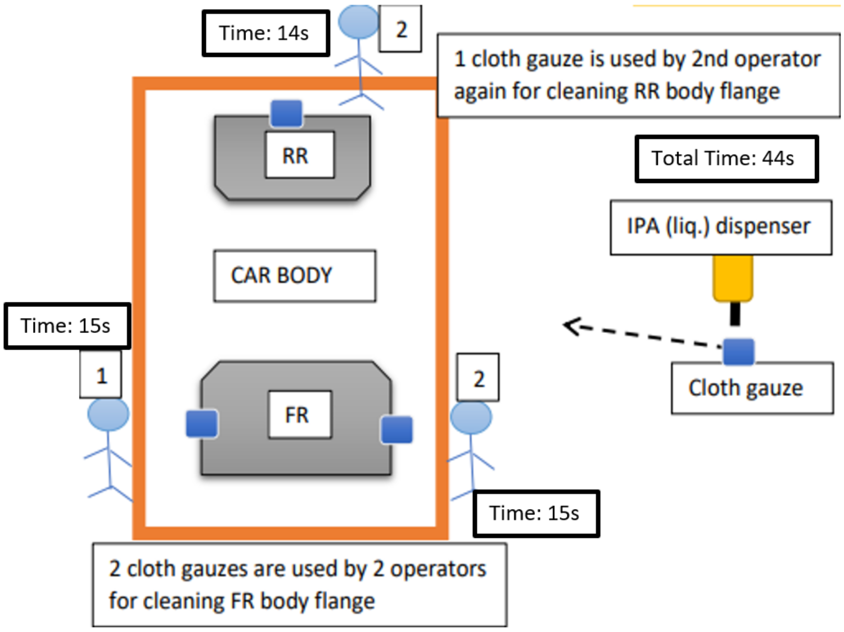


**Figure 1.7:** Visualisation of the complete degreasing process done manually by 2 human operators using cloth and liquid IPA.[65]

## Change of cleaning/degreasing materials

While finding the automation solution, the author, being an engineer in the Assembly Engineering team of TMMCZ, has also proposed a new wiping material for this process by replacing the two cleaning agents (**liquid IPA and cleaning cloth/wipe**) with one cleaning agent (**industrial pre-saturated IPA wet wipes with 99 percent IPA concentration**). This helps in eliminating **one extra step** of wetting the cloth/wipe with liquid IPA, while also reducing material costs, safety risks of liquid alcohol for human operators and also, saving storage space and overall waste.[65] This doesn't affect the previous process time (44s) or any other significant change in the old process. (see the visualisation of degreasing process with new cleaning material in figure 1.8)

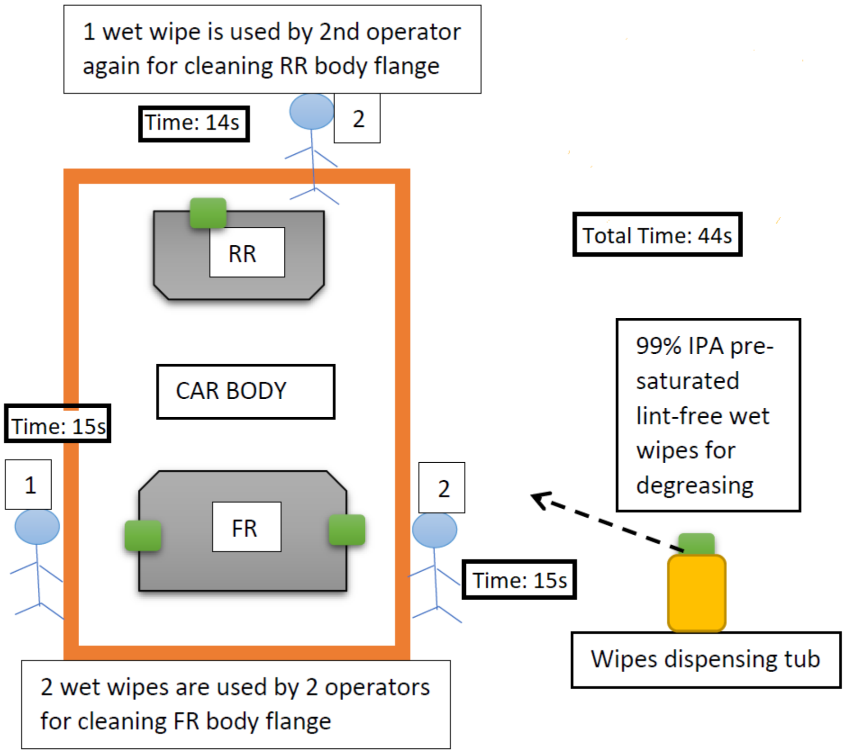


**Figure 1.8:** Visualisation of the complete degreasing process done manually by 2 human operators using wet wipes.[65]

- These two choices for cleaning agents will also be available for the automated solution (to be proposed in the next chapters) to choose from, as the automated gripping or picking tool will have to be programmed as to choose the cloth to wet with liquid IPA or just pick up the wet wipe to directly do the wiping application.

### 1.1.3 Problems arising from current degreasing process

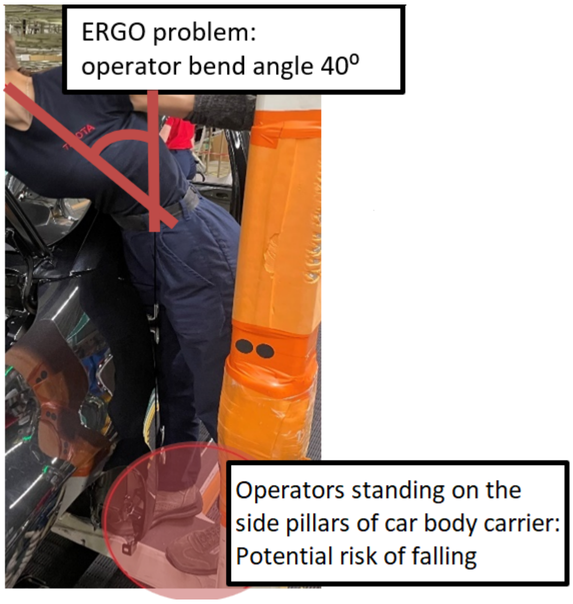


**Figure 1.9:** Ergonomic problems visualisation.[57]

- **Ergonomics:** There is a clear issue with the operators' ergonomics regarding their bending angle during the wiping of the body flange (see figure 1.9). The average angle of bending is 40 degrees which is much higher than the ideal bending angle of 20 degrees (see figure 1.10) as recommended by Process Design Standards from TMMCZ.

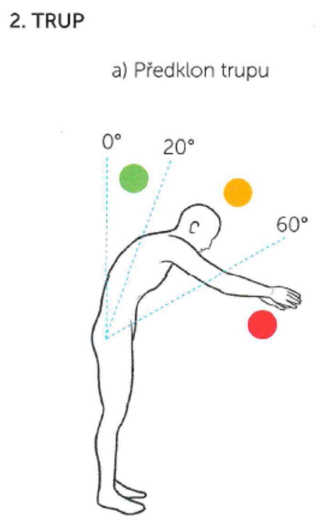


**Figure 1.10:** Ideal ergonomic bending angle for the operators.[61]

- **Risk of Physical Injury:** As the operators stand on the side pillars of the car body carriers in continuous motion in the Assembly line, there is a huge risk of operators to slip or fall down from the car body carriers. There have been similar incidents in the past which makes this a serious concern (see figure 1.9).

- **Health Risks:**
  1. Directly working with *99 percent liquid isopropyl alcohol* is medically dangerous for human operators for a long term continuous use, as the long term contamination from smell and touch while using the liquid dispenser (see section 1.1.2) can cause irritation in skin and eyes of the operators working with the liquid alcohol.[6]
  2. Also, the ventilation in the workplace isn't adequate enough due to this workplace being used for different process in the past which now changed to degreasing process.
- **Quality of Process:** Degreasing process, being manually done, brings a margin of potential human errors in the application.
  1. Firstly, there seems to be a potential inconsistency of the amount of alcohol being dispensed to wet the cloth piece due to rotation of operators' shifts, meaning every operator may use different amount of alcohol to wet the cloth.
  2. Secondly, the wiping of the surface is potentially bound to human errors as 2 operators wipe the front windshield body flange, meaning there may be no consistency in the start and end position of the wiping on the body flange surface on every car body.
  3. Third factor is the contact force applied by different operators while wiping the body flange, it can be different on every car body due to their physical differences (height, physique).

  So, these three potential problems may lead to every car body being wiped differently with seemingly no consistency in the overall quality of cleaning/degreasing of the body flange, this can lead to potential issues in the later process of glass installation (see in section 1.1.2).
- **Process time:** Again, human error can lead to inconsistency in process time for degreasing of different car bodies deviating from the standard 44 seconds (section 1.1.2).
- **Need for Special Storage:** Due to the inflammable nature of liquid isopropyl alcohol, there is additional need for a special storage/cabinet suitable for storing such dangerous inflammable goods.
- **Need for Special Waste Disposal:** After being wetted with liquid IPA alcohol and being used for wiping, the used cloth pieces need to be disposed of in the same way as other dangerous goods.
- **Residue:** Using cloth pieces for wiping can leave some residue on the car body flange surface.

### Problems left after changing the cleaning/degreasing materials

As discussed in the subsection 1.1.2, the current cleaning/degreasing materials (liquid IPA and cloth pieces) will be replaced with pre-saturated IPA wet wipes (Industrial degreasing wet wipes), so now another review of the problems with the degreasing application:-

- **Risk of Physical Injury** still exists as the application process is same as before.
- **Health Risks** are reduced as the pre-saturated IPA wet wipes are less contaminated and dangerous than liquid IPA alcohol. But the risks as mentioned before, although a bit reduced, still exist.
- **Quality of Process** is mostly affected in the same way as mentioned before, with the exception of inconsistency of amount of liquid alcohol being dispensed as the wet wipes have exact same concentration of IPA alcohol in each wet wipe. But the manual application still brings the same human errors as before.
- **Process Time** is still unchanged.
- **Need for Special Storage** is reduced as there is only one cleaning/degreasing material instead of two.
- **Need for Special Waste Disposal** is reduced as the wet wipes dry much faster than normal wetted cloth, thus, can be disposed of as a normal dry wipe.
- **Residue** is completely solved as the industrial wet wipes are completely lint-free or residue-free.

## 1.2 First thoughts for finding solution

After understanding the crux of the problems brought forward in the subsections 1.1.3 and 1.1.3, two major issues come out - operators' safety risks and overall quality of cleaning/degreasing. Based on this observation, the author has proposed to find an automated solution for this application, so as to move the operators away from this process to other less risky processes instead and also, to maintain a consistency in the quality of work along with many other benefits which will be studied in the subsection 3.1.1. The author will study different automated solutions and propose the optimum solution to design and simulate, leading to the objective of this thesis.

# Chapter 2

## Objective of the Thesis

After understanding the current situation with the workplace and current process application in the previous chapter 1, the author aims to make a proposal of an automated solution for the given process of car body flange degreasing at TMMCZ Kolin which can later be given to the automation technology partner/supplier of TMMCZ for possible real-time implementation.

The objective to propose such a solution, thus, will be divided into four parts:

1. **Study of the concepts of automation and industrial automation technologies:**
   - Understanding the concepts and need of automation and how it leads to industrial automation.
   - Understanding the concepts of Industry 4.0 and its components to apply that to integrate the given process with automation.
   - Understanding the industrial automation technologies available in the market.
2. **Comparison of different automated solutions to find the optimal solution to continue with:**
   - Trying different automated solutions using the technologies available.
   - Comparing the different solutions based on factors necessary for possible real-time integration into the TMMCZ factory.
   - Choosing the optimal solution.
3. **Design of the chosen automation solution:**
   - Designing the complete configuration of the chosen automation technologies with the car model and car body carrier.
   - Designing the possible gripper prototype.
   - Describing the complete automated degreasing process.

4. **Simulation of the solution and programming:**
   - Simulation of the complete designed configuration of the chosen automation technologies and car model using RoboDK software.
   - using RoboDK to generate programs for the automation solution.

# Chapter 3

# Overview of concepts

Automation is the development and application of technologies that allow goods and services to be produced and delivered with little or no human interaction. Many tasks that were previously performed by humans are now more efficient, reliable, and/or quick thanks to the use of automation technologies, techniques, and processes.[52]

Manufacturing, transportation, utilities, defense, facilities, operations, and, more recently, information technology are all using automation.[52]

***Why Automation?*** Automation is usually used to reduce labor costs or to replace humans in most menial or repetitive tasks. Automation can be found in almost all industries and niches, but it is most prominent in manufacturing, utilities, transportation, and security.[52]

Most industrial firms, for example, use robotic assembly lines as part of their automated processes. Human involvement is only necessary to define and manage operations; the assembly of the various components is left to the machines, which convert raw materials into completed things automatically.[52] From installation, integration, and maintenance to design, procurement, and administration, automation touches every aspect of industry. These sectors' marketing and sales functions are also impacted by automation.[53]

Robotics and expert systems, telemetry and communications, electro-optics, Cybersecurity, process measurement and control, sensors, wireless applications, systems integration, test measurement, and many more technologies are all part of automation.[53]

## 3.1 Industrial Automation

Industrial automation is a core aspect of the industry in the world today. It is what makes today's manufacturing capabilities so advanced with higher efficiency, speed, precision, and accuracy while also significantly reducing the usual industrial human repetitive tasks and safety risks.

Manufacturing companies urgently require production systems with the flexibility to enable dynamic product diversity as well as reconfigurability to create new products economically. A new collaborative automation paradigm is developing in answer to this need, which is defined by decentralized distributed automation systems made up of intelligent re-configurable mechatronics modules.[59]

Industrial automation is the use of various control devices, such as PCs, PLCs, and DCS, to automate various operations in an industry without requiring considerable human intervention and to deliver automatic control performance. Control techniques in industries employ a set of technologies that are deployed to achieve the intended performance or output, making the automation system significant for industries.[29]

Industrial automation requires the use of advanced control strategies such as cascade controls, modern control hardware devices such as PLCs, sensors and other instruments for sensing control variables, signal conditioning equipment to connect the signals to the control devices, drives and other significant final control devices, standalone computing systems, communication systems, alarming, and HMI (Human Machine Interface) systems.[29]

### 3.1.1 Need for Industrial Automation

- **To reduce the need for periodic or manual checking:** In some critical applications, it is necessary to perform industrial operations on a regular basis by checking the process variable. Automation equipment reduces the number of periodic or manual operations and creates automatic working conditions.[29]

- **To increase the Productivity:** Manufacturing and other production processes when automated leads to increase the production rates by producing more output for a given labor input.[29]

- **To reduce the Production Cost:** With the use of automated machines and equipment, human intervention in process control is reduced significantly. This reduces the investment in labor costs and, as a result, the production cost.[29]

- **To improve product Quality:** Continuously performing the same task may not be perfect in all cases in terms of quality specifications with using human efforts. By utilizing real-time hardware control devices, automation equipment can provide consistent and reliable product quality.[29]

- **To increase the Process Flexibility:** Using automation equipment, various processes are handled simply and without creating a complex environment, which is especially important in manufacturing processes.[29]

- **To make the process more Operator-Friendly and to improve the Safety Risks:** The complexity of operating the equipment or processes is reduced by industrial automation. It elevates the operator's position from that of operator to that of supervisor.[29]

### 3.1.2 Types of Industrial Automation

Based on the flexibility and level of integration in manufacturing processes, automation systems are classified into four basic types:

1. **Fixed Automation:** The equipment configuration determines the sequence of operations to be performed. It is used in high-volume production with specialized equipment. Automated assembly lines, distilled processes, and machine transfer lines are examples of this automation system.[29]

**Figure 3.1:** Fixed Automation.[29]

2. **Programmable Automation:** The sequence of operations can be changed by changing the program. The order of operations varies depending on the product configuration. New programs can also be entered into programmable devices for new products. This type of system is used in batch processes, steel rolling mills, industrial robots, and other similar applications.[29]

**Figure 3.2:** Programmable Automation.[29]

3. **Flexible Automation:** It is a programmable automation extension. This provides more flexibility in dealing with product design variations. If operators want to change the sequence of the process, they can give commands in the form of codes in the computer program. Lower level equipment is given instructions to operate in the field without wasting production time. This type of automation is used in the production of multipurpose CNC machines, automatic guided vehicles, etc.[29]

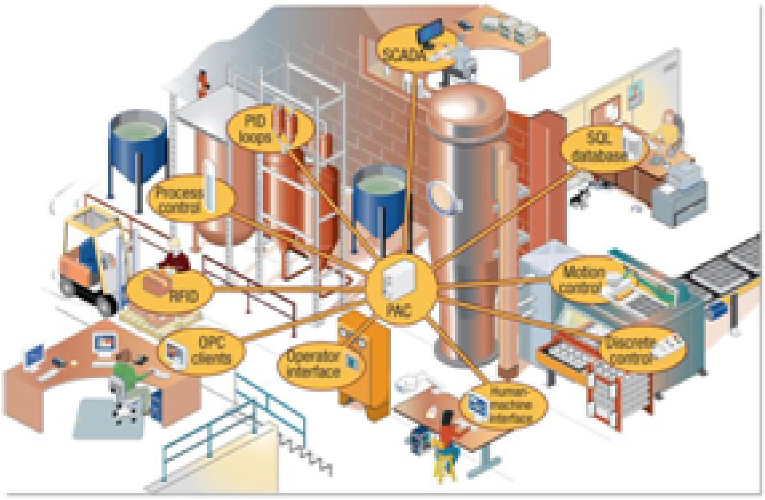


**Figure 3.3:** Flexible Automation.[29]

4. **Integrated Automation:** The entire system is fully automated and controlled by a computer. The entire system is completely automated, from the design process to the dispatching. The robots even handle the equipment. This system is employed in computer-integrated manufacturing systems.[29] One example is shown in Figure 3.4.

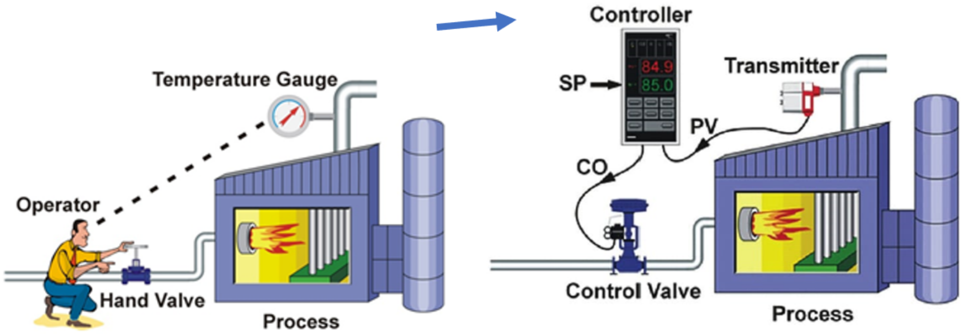


**Figure 3.4:** An example showing integrated automation of the manual control of water temperature using PID controller and electronic temperature measurement device.[26]

### 3.1.3 Automation Equipments

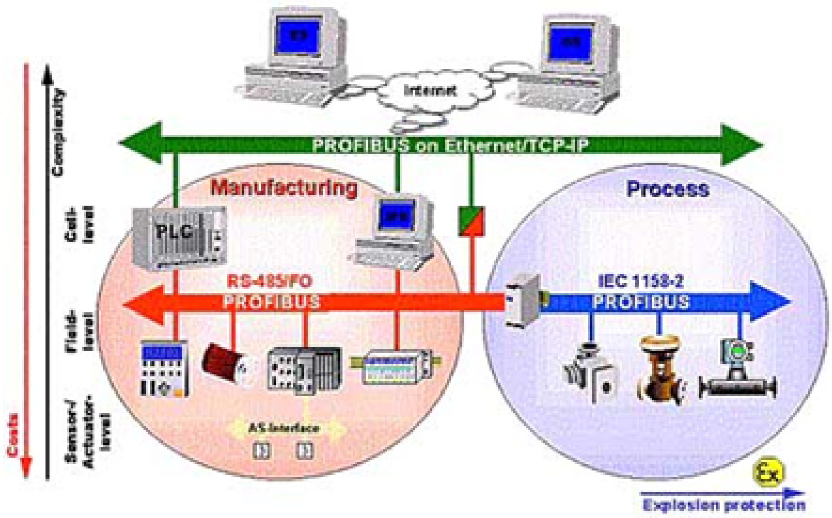


**Figure 3.5:** Equipments of Automated Industry.[29]

- **Sensors and Actuators:** A sensor detects and converts various process variables into electrical or optical signals. Temperature, pressure, velocity, and flow are examples of these sensors. To gain control over processes, actuators convert electrical signals to mechanical means. Relays, magnets, servomotors, etc. fall into this category. Some sensors and actuators can communicate with industrial field communication buses, which fall under the category of smart devices.[29]

- **Industrial Controllers:** Programmable logic controllers (PLCs), also known as industrial controllers, can be programmed to perform specific control functions. It is made up of a CPU or processor, I/O modules (both analog and digital), and relay modules that connect the various input/output devices. These can be modular, which is a fixed type, or integrated, which allows modules to be extended based on the inputs available. In addition to PLCs, conventional PCs are used to control the process online or by changing the programs. PLCs include dedicated software for programming the control strategy.[29]

- **HMI (Human Machine Interface):** HMIs include features such as displaying information on computer screens and other displays, logging results in a database, and sending out alarm signals. It makes use of visual-based technologies such as SCADA (Supervisory Control and Data Acquisition).[29]

- **Communication System:** Many sensors, actuators, controlling PCs, and other control devices are geographically distributed in industries and communicate with each other via multiple data buses. Factory buses, Process buses, and Field buses are the three types of buses used in industrial automation. The field bus communicates with field instruments and control devices, while the process bus links supervisory level computers to control devices such as PLCs. The factory bus connects the organization's higher levels to the supervisory level. RS-485, ProfiBus, CAN control ModBus, etc. are used for communication.[29]

## 3.2 Industry 4.0 - the Fourth Industrial Revolution

Due to increased inter-connectivity and smart automation, the Fourth Industrial Revolution, also known as Industry 4.0 or 4IR, envisions substantial change in technology, industries, and societal patterns and processes in the 21st century.

The world's industrialized countries are currently proceeding through the fourth stage of industrialization, known as Industry 4.0. It was followed by the third stage of the industrial revolution, which began in the early 1970s and was based on automation enabled by information technologies.[72]

**Quick overview of the four industrial Revolutions:-**

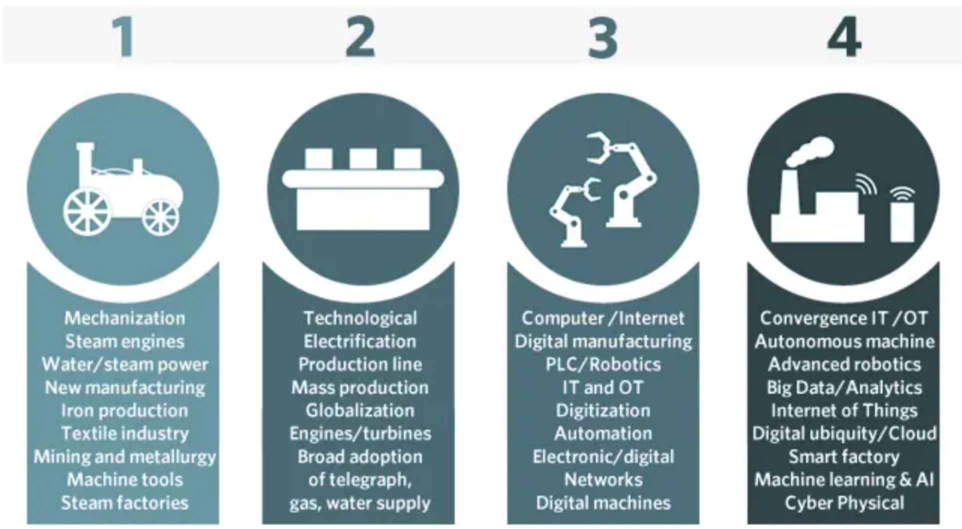


**Figure 3.6:** Industry 1.0 to 4.0.[17]

1. **First Industrial Revolution:** led to the invention of steam machines to enable usage of water and steam power which would then result in the industrial transformation of the society with the development of trains and mechanisation of manufacturing.[17]

2. **Second Industrial Revolution:** led to the usage of electricity to enable new manufacturing inventions such as the assembly line which

then led to the development of mass production and to some extent, of automation as well.[17]

3. **Third Industrial Revolution:** the period of the rise of computers, computer networks (WAN, LAN, MAN,..), the rise of robotics in manufacturing, connectivity and the birth of Internet, changing the way information is handled and shared, thus, leading to further automation with the new electronic versions of many processes and physical environments.[17]

4. **Fourth Industrial Revolution:** moving from the Internet and client-server model to ubiquitous mobility, the bridging of digital and physical environments, the convergence of IT (Information Technology) and OT (Operational Technology), along with the advancement brought by advanced robotics and AI/cognitive, enabling Industry 4.0 with automation and optimisation in whole new ways to innovate and truly fully automate the industry and bring it to the next level.[17]

Industry 4.0 offers a significant advantage in terms of accelerating manufacturing while lowering costs and increasing productivity. Smart manufacturing, smart engineering, smart energy, and smart logistics are all part of the smart factory environment.[64]

### 3.2.1 Key Elements of Industry 4.0

CPS (cyber-physical systems), IoT (Internet of Things), Cloud computing, and Cognitive computing are key elements of Industry 4.0. When these four elements are integrated and they interact with a factory's physical methods to generate a smart environment, the factory is referred to as a smart factory.[63]

1. A **CPS (cyber-physical system)** is a system that combines physical and software components and is controlled by computer-based algorithms. It allows users and physical systems to collaborate in a smart environment to generate desired output with less input. 12 Robotics systems, autonomous automotive systems, smart grid, process control systems, medical monitoring, and automatic pilot avionics are all applications of CPS.[69, 67]

2. The **IoT (Internet of Things)** entails extending internet connectivity to devices such as laptops, computers, tablets, mobile phones, and other similar devices at any location around the world in order to connect, interact, and exchange data with users, as well as physical systems to enable remote monitoring and control of physical systems.[54]

3. **Cloud computing** refers to shared pools of configurable computer system resources and higher-level services that may be deployed quickly and with minimal administration effort, usually over the Internet. Cloud

computing, like a public utility, relies on resource sharing to achieve coherence and economies of scale.[2]

4. **Cognitive computing** is a new type of computing that aims to create more accurate models of how people react to inputs. It employs artificial intelligence and signal processing techniques. Among the technologies included in these platforms are reasoning, natural language processing, machine learning, vision (object recognition), speech recognition, dialog, and narrative generation, as well as human–computer interaction.[66]

An example integrating all these elements of Industry 4.0 to make a Smart Factory is shown in the figure:

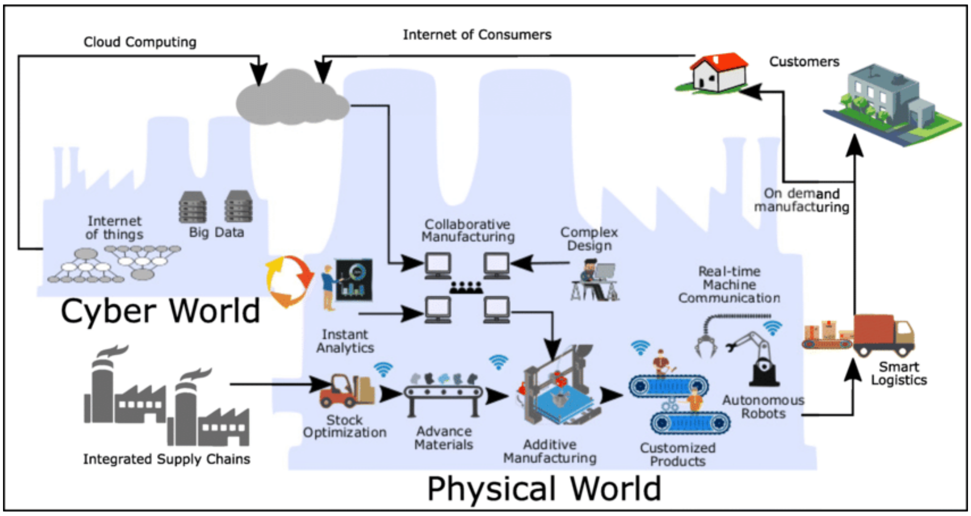


**Figure 3.7:** Integration of Industry 4.0 elements.[58]

## 3.3 Automation technologies used in Industries

Overview of automation technologies used in modern manufacturing industries, some of which will be considered for proposing the optimal automated solution in the next chapter 4:

### 3.3.1 Linear Motion Units:

Linear technology is one of the most widely used technologies in industry when it comes to automation. It's robust, reliable, and easy to implement. What makes this technology so popular in industrial usage, it is the movement of this system which always ravels along an axis. The object to be moved is carried on a **slide** (see figure 3.8) that travels forward and back along the axis.[25]

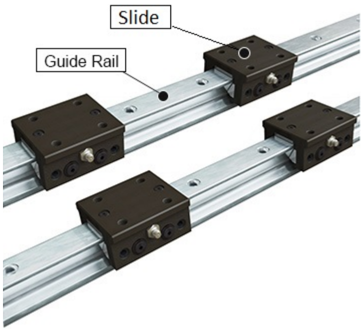


**Figure 3.8:** Linear guide rail with *Slide*.[40]

Linear units can also be combined, allowing for two- and three-dimensional positioning. This method can be used to execute complex sequences at a low cost. The linear guide, a crucial element in any linear unit, is responsible for determining the direction of motion and consists of two parts – the guide (rail) and corresponding transport element (slide) (see figure 3.8).[25]

## Key components of a Linear Unit

A linear unit has two key components:-

1. **Linear Guide** there are three types of linear guides:

   a. ***Roller units:*** Rollers that run along cylindrical shafts are used in this type of linear guide. The rollers' design support play-free travel and prevents them from jumping out of their guide. **Advantage**: Roller units offer a very high level of flexibility.[25]

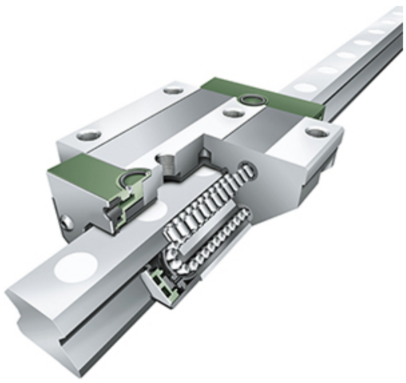

**Figure 3.9:** Cutaway view of roller guide.[37]

   b. ***T-slot sliders:*** A profiled slider runs along a profiled rail in these systems. Because of this profiling, the slide is unable to exit the guide track. **Advantage:** T-slot sliders are ideal for applications that do not require complete freedom of movement and are generally recommended when working on a limited budget.[25]

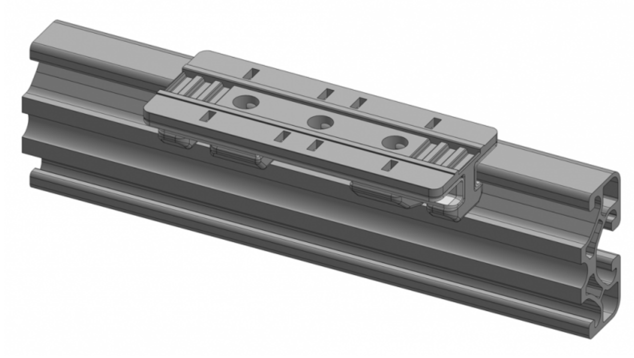

**Figure 3.10:** Side profile view of t-slot slider.[43]

c. ***Recirculating ball bearing units:*** The ability of these systems to accommodate high loads in a small space is their defining feature. The rolling elements are arranged in the multiple points of contact principle so that they run along one line before being recirculated along another. **Advantage:** This system has a low friction coefficient and excellent load distribution.[25]

**Figure 3.11:** Internal view of recirculating ball bearing guide.[24]

2. **Drive technologies in linear unit** Modern drive technologies are always tailored to specific tasks. They can be either incredibly fast or incredibly accurate. The overall performance of a linear unit is primarily determined by the drive technology used, which has a significant impact on the precision, speed, carrying capacity, and costs of a linear solution.[25] There are primarily four types of drive technologies available:

a. ***Timing-belt drive:*** This technology employs a toothed drive belt that mechanically locks around a toothed pulley driven by a motor. This mechanical interlocking eliminates slip and ensures the transmission of high forces. A timing belt is made up of steel cables encased in a polyurethane sheath, has a long service life, and allows for a smooth running action. **Advantage:** this technology is that it allows for extremely dynamic movements and, as a result, short cycle times.[25]

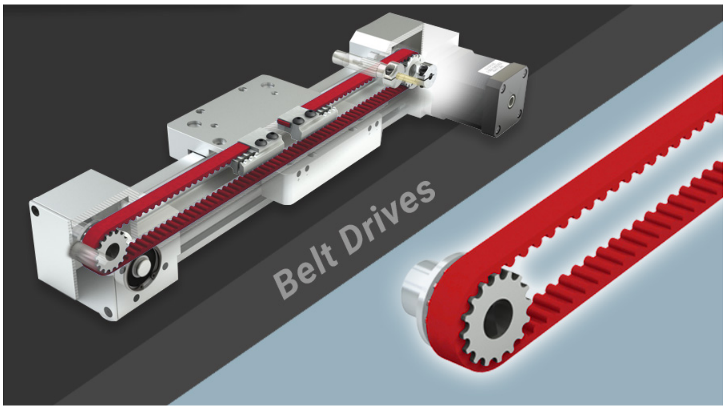


**Figure 3.12:** Internal view of timing-belt drive.[3]

b. ***Ball/lead screw unit:*** A precision spindle serves as the basis of a Ball Screw Unit. The lead on the thread greatly influences the system's speed and positioning accuracy. The spindle is equipped with a non-turning drive nut that houses ball bearings. As the spindle turns, these ball bearings circulate in the thread, ensuring that the nut moves along a straight axis. Because the ball bearings are slightly larger than the track on which they run, they create a pre-tensioning effect that eliminates play and increases load carrying capacity. **Advantage:** Ball Screw Units are ideal for precise positioning and high power.[25]

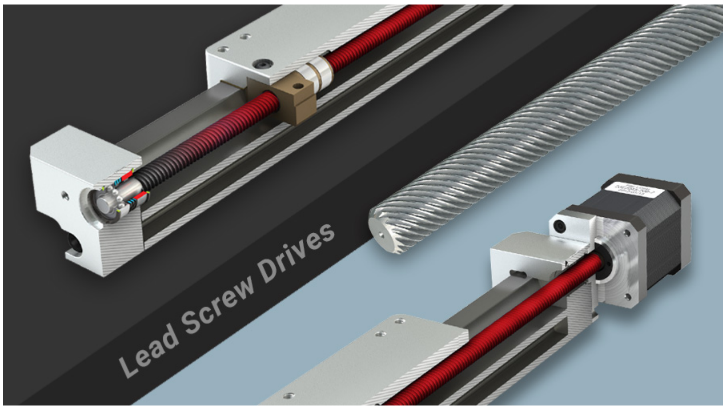


**Figure 3.13:** Internal view of ball/lead screw unit.[3]

c. ***Chain drive:*** Linear units with chain drives transfer large forces in the direction of travel, but their design limits their positioning and travel speed. However, because of their high failure load, chain drives are frequently used in vertical applications. **Advantage:** Chain drives are resistant to soiling problems, can transfer high forces, and are ideal for vertical movements. Because the force in a chain drive can be converted into a rotary motion via sprocket wheels positioned anywhere on the linear unit, this design is particularly well suited to building conveyor systems with rollers.[25]

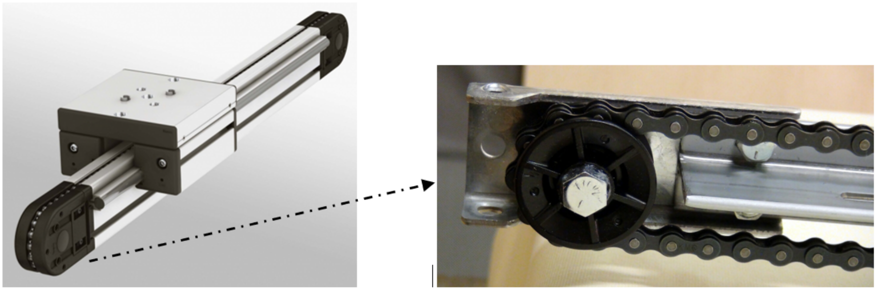

**Figure 3.14:** Chain drive and its internal view.[31, 23]

d. ***Rack drive:*** In a rack drive, the driven gearwheel interlocks with the straight rack to eliminate the possibility of slip. Thus, the rotary motion of the drive motor is directly converted into the rectilinear motion of the slide. This allows for two applications:

(i) The load follows the driven gearwheel.

(ii) The drive is fixed in place, and the load moves along with the rack.

**Advantage:** A rack drive is strong, can safely lift heavy loads, and can accurately position devices over long distances.[25]

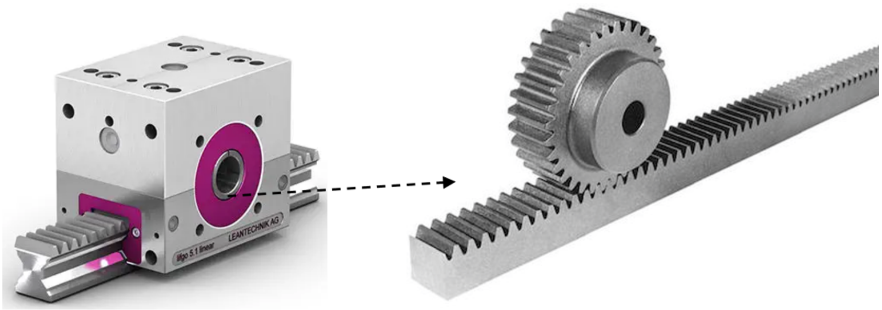

**Figure 3.15:** Rack drive and its internal view.[32, 33]

## Applications of linear technology

The linear units can be combined together along with some other technological components and tools for various applications. Some of the commonly used applications (to be considered by the author to find the automation solution in the next chapter 4) are:-

1. ***Linear unit (single axis):*** As part of a process automation solution, a Linear Unit is used to move workpieces, tools, or a robot along a line

to execute screwing, drilling, or joining applications. The length of a single profile does not limit the length of the linear axis. Many models allow the user to extend the line which the Linear Unit travels along. All Linear Units can be configured to meet the needs of the task at hand. Motors can be connected with carefully configured drive components.[18]

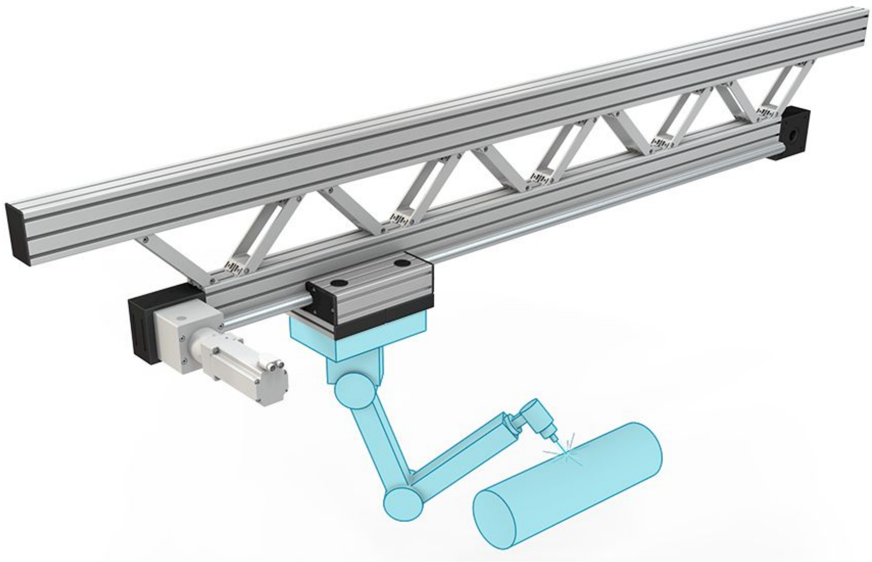

**Figure 3.16:** Linear unit in single axis with a robot.[18]

2. ***2D gantry:*** 2D gantries are ideal for automation tasks that require a tool to travel to multiple points in a single plane using linear automation technology. These two-axis solutions are used, among other things, to print or inspect surfaces.[18]

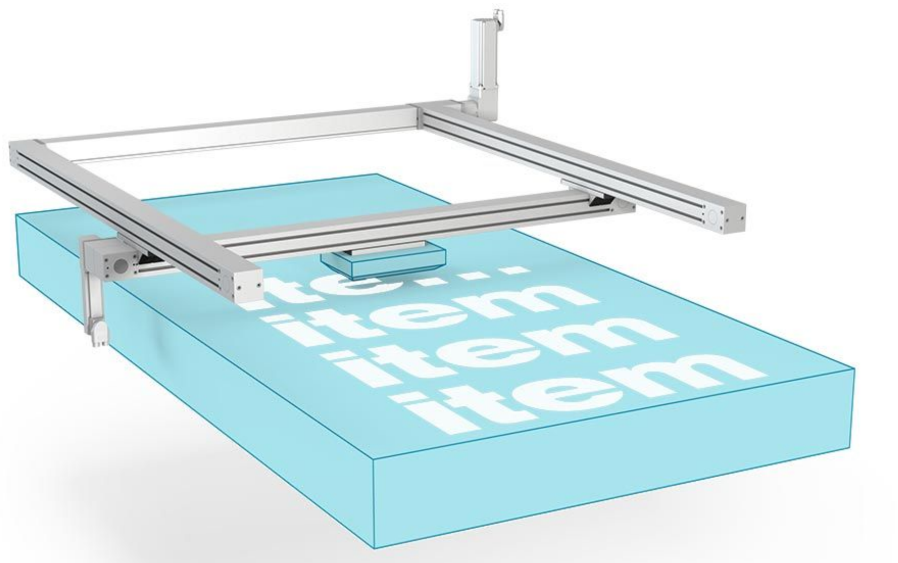


**Figure 3.17:** Linear units combined to form 2D gantry.[18]

3. ***3D gantry:*** 3D gantries perform difficult three-dimensional positioning tasks. They are used to lift, move, and place workpieces for stacking, sorting, and machining operations. The grippers and tools for the task can be fitted to the item carriages and support profiles when designing pick-and-place systems.[18]

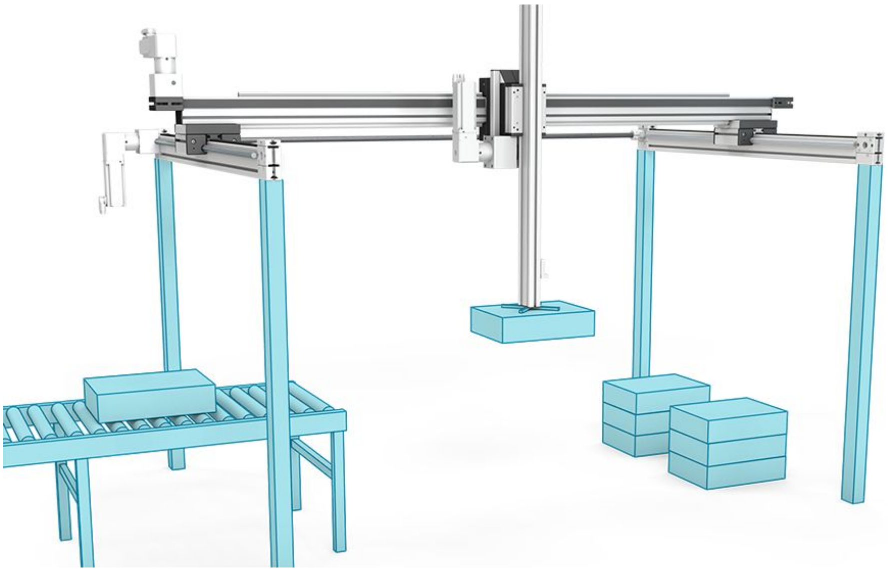

**Figure 3.18:** Linear units combined to form 3D gantry.[18]

4. ***Cantilever axis:*** Cantilever axes use the Linear Unit's support profile to exert force on a workpiece or precisely position a tool. They are well suited for a wide range of production and logistics applications, as well as material and load testing.[18]

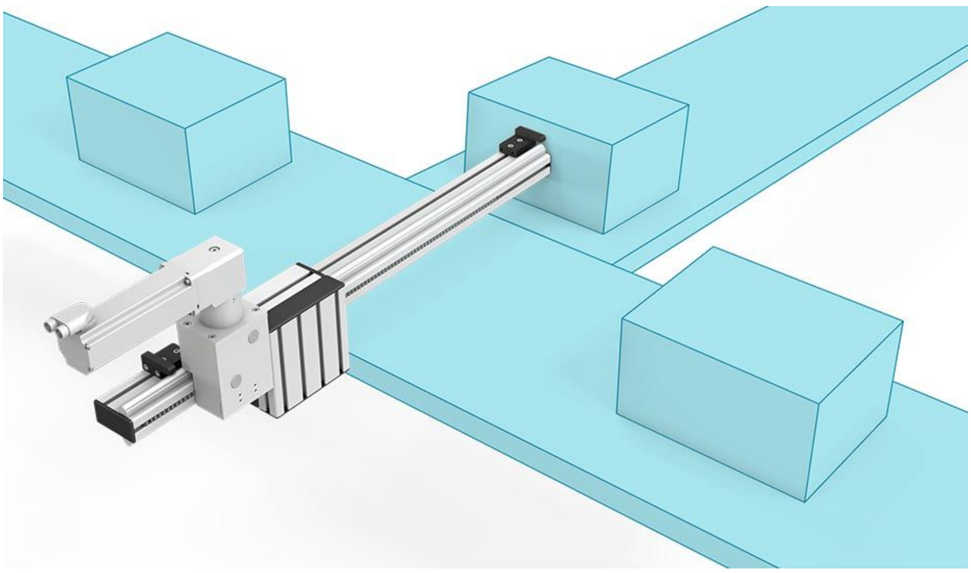

**Figure 3.19:** Linear units as cantilever axis to push boxes.[18]

5. ***XY axis table handling:*** In XY tables, grippers, suction cups, lasers, and scanners are used for machining, sorting, positioning, and inspecting workpieces. This type of 2D linear gantry is ideal for dynamic pick-and-place tasks.[18]

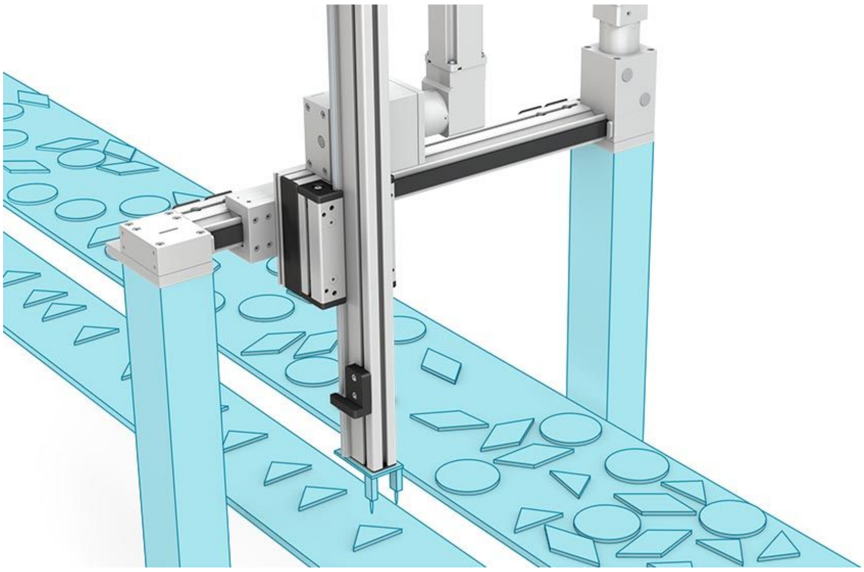

**Figure 3.20:** XY axis gantry for pick-and-place operation.[18]

### 3.3.2 Industrial Robots:

An industrial robot is a manufacturing robot system. Industrial robots are fully automated, programmable, and can move on three or more axes.[15]

Robots are commonly applicable for welding, painting, assembly, disassembly, pick and place for printed circuit boards, packaging and labeling, palletizing, product inspection, and testing, all of which require a high level of endurance, speed, and precision. They also can help with material handling.[15]

Industrial robots are increasingly being used in a wide range of industries and applications because they can be programmed to perform dangerous, dirty, and/or repetitive tasks with consistent precision and accuracy. They are available in a variety of models, with the most common distinguishing features being the reach distance, payload capacity, and the number of axes of travel (up to six) of their jointed arm.[16]

A robot uses an **end effector** or **end of arm tooling (EOAT)** (also known as **gripper** or **gripping tool**) attachment to hold and manipulate either the tool performing the process or the piece upon which the process is being performed in both production and handling applications.[16]

The actions of the robot are instructed by a combination of programming software and controls. Their automated functionality enables them to operate around the clock and on weekends, as well as with hazardous materials and in difficult environments, freeing up personnel for other tasks. Robotic technology also boosts productivity and profitability while removing labor-intensive tasks that could cause physical strain or injury to workers.[16]

## Types of Industrial robots

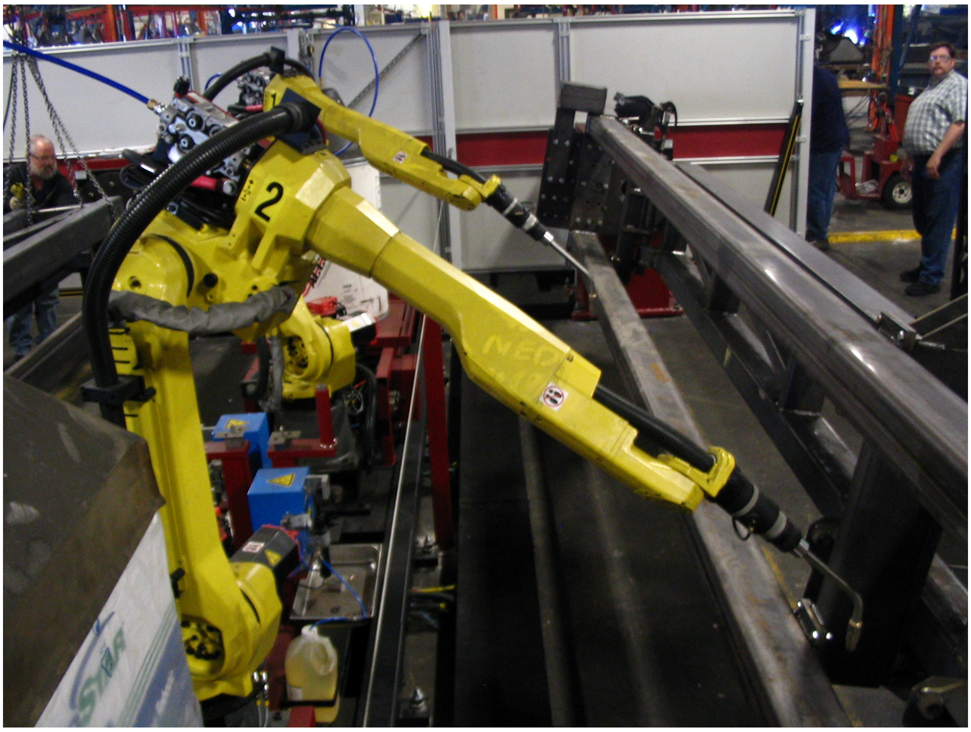

**Figure 3.21:** A set of 6-axis robots being used for welding operations.[15]

There are primarily six types of industrial robots:-

1. ***Articulated robots:*** The most common industrial robots are articulated robots. They have the appearance of a human arm, which is why they are also known as robotic arms or manipulator arms. Their multiple degrees of freedom articulations allow the articulated arms to move in a variety of ways.[15]

2. ***Cartesian coordinate robots:*** Cartesian robots, also known as rectilinear, gantry, or x-y-z robots, have three prismatic joints for tool movement and three rotary joints for tool orientation in space. A six-axis robot (see figure 3.21) is required to move and orient the effector organ in all directions (or degrees of freedom). Three axes are sufficient in a two-dimensional environment, two for displacement and one for orientation.[15]

3. ***Cylindrical coordinate robots:*** Cylindrical coordinate robots are distinguished by a rotary joint at the base and at least one prismatic joint connecting their links. They can slide vertically and horizontally. The compact effector design enables the robot to reach tight workspaces without sacrificing speed.[15]

4. ***Spherical coordinate robots:*** Rotary joints are the only type of joint found in spherical coordinate robots. They were among the first robots to be used in industrial applications. They are commonly used for die-casting machine tending, plastic injection and extrusion, and welding.[15]

5. ***SCARA robots:*** Selective Compliance Assembly Robot Arm (SCARA) is an abbreviation for Selective Compliance Assembly Robot Arm. SCARA robots are distinguished by their two parallel joints that allow movement in the X-Y plane. At the effector, rotating shafts are positioned vertically. SCARA robots are used in jobs that necessitate precise lateral movements. They are ideal for assembly work.[15]

6. ***Delta robots:*** Parallel link robots are another name for delta robots. They are made up of parallel links that connect to a common base. Delta robots are particularly useful for tasks requiring direct control and high maneuverability (such as quick pick-and-place tasks). Delta robots use four bar or parallelogram linkage systems.[15]

### Defining parameters for robots

1. ***Number of axes:*** to reach any point in a plane, two axes are required; to reach any point in space, three axes are required. Three more axes are required to fully control the orientation of the end of the arm (griper or EOAT). Some designs (for example, the SCARA robot) trade motion limitations for cost, speed, and accuracy.[15]

2. ***Degrees of freedom:*** typically, this is the same as the number of axes.[15]

3. ***Working envelope:*** the area of space that a robot can traverse.[15]

4. ***Kinematics:*** the actual configuration of rigid members and joints in the robot that determines the robot's possible motions Robot kinematics classes include articulated, Cartesian, parallel, and SCARA.[15]

5. ***Payload or carrying capacity:*** the amount of weight that a robot can lift.[15]

6. ***Speed:*** how quickly the robot can move the end of its arm This can be expressed in terms of the angular or linear speed of each axis, or as a compound speed, i.e. the speed of the arm's end when all axes are moving.[15]

7. ***Acceleration:*** how fast an axis can accelerate. As this is a limiting factor, a robot may be unable to reach its specified maximum speed for short-distance movements or complex paths requiring frequent changes of direction.[15]

8. ***Accuracy:*** how close a robot is able to come to a commanded position The error is a measure of accuracy when the absolute position of the robot is measured and compared to the commanded position. External sensing, such as a vision system, infrared or 3D camera systems, can improve accuracy. Accuracy can vary depending on speed, position within the working envelope, and payload.[15]

9. ***Repeatability:*** how well the robot will return to a predetermined position This is not the same as accuracy. It's possible that when told to go to a specific X-Y-Z position, it only gets to within 1 mm of that position. This is its accuracy, which can be improved through calibration. However, if that position is taught into controller memory and each time it is sent there, it returns to within 0.1mm of the taught position, then the repeatability is within 0.1mm.[15]

10. ***Motion control:*** for some applications, such as simple pick-and-place assembly, the robot only needs to return to a small number of pre-taught positions repeatedly. Motion must be continuously controlled to follow a path in space, with controlled orientation and velocity, for more sophisticated applications such as welding and finishing (spray painting).[15]

11. ***Power source:*** some robots are powered by electric motors, while others are powered by hydraulic actuators. The former is faster, while the latter is stronger and more useful in applications such as spray painting.[15]

12. ***Drive:*** some robots use gears to connect electric motors to joints, while others connect the motor directly to the joint (direct drive).[15]

13. ***Compliance:*** this is a measurement of how far or how far an axis of a robot will move when a force is applied to it. Because of compliance, when a robot moves to a position with its maximum payload, it will be slightly lower than when it moves to a position with no payload. Overshoot can also be caused by compliance when carrying heavy payloads, in which case acceleration must be reduced.[15]

## Benefits of industrial robots

1. ***Accuracy:*** robots can be programmed to do the repetitive tasks with precision. For example, robotic palletizers can be guided by software to ensure proper load placement.[16]

2. ***Flexibility:*** robotic systems can be re-purposed for other purposes, and end effectors can be swapped out to handle different types of loads.[16]

3. ***Lower labour costs:*** robotised processes especially for repetitive tasks reduce worker strain and free the operators for other supervisory tasks.[16]

4. ***Quiet operation:*** servo-based robots generate low noise levels during their operation.[16]

5. ***Reduced product damage:*** robots working directly with the products in many manufacturing industries are configured for gently handling to prevent package and product damage.[16]

6. ***Speed:*** depending on the process or task assigned, robotic systems can increase productivity rates up to 50 percent.[16]

### End-of-arm-tooling (EOAT)

The end effector, also known as end-of-arm-tooling, is the most important robot peripheral (EOAT). Welding devices (such as MIG-welding guns, spot-welders, and so on), spray guns, and grinding and deburring devices (such as pneumatic disk or belt grinders, burrs, and so on) are common examples of end effectors (devices that can grasp an object, usually electro-mechanical or pneumatic). Objects can also be picked up using a vacuum or magnets. End effectors are frequently highly complex, designed to match the handled product, and frequently capable of picking up a variety of products at once. They may use a variety of sensors to help the robot system locate, handle, and position products.[15]

### Robot programming

An industrial robot's motions and sequences are typically taught by connecting the robot controller to a laptop, desktop computer, or (internal or Internet) network.[15]

A work-cell, or cell, is a collection of machines or peripherals that includes a robot. A typical cell might include a parts feeder as well as a robot. The various machines are 'integrated,' meaning they are controlled by a single computer or PLC. The robot's interactions with other machines in the cell must be programmed, both in terms of their positions in the cell and synchronization.[15]

- **Software** The corresponding interface software is installed on the computer. The use of a computer simplifies the programming process significantly. Depending on the system design, specialized robot software is run in either the robot controller or the computer, or both. Positional data and procedure are the two fundamental entities that must be taught (or programmed). The robot software's purpose is to make both of these programming tasks easier.[15]Each manufacturer has their own robot software. While the vast majority of software is about manipulation of data and seeing the result on-screen, robot software is for the manipulation of objects or tools in the real world.[35]

- **Positional commands** The robot can be guided to the required position using a graphical user interface (GUI) or text-based commands in which the required X-Y-Z position can be specified and edited.[15]

- **Teach pendant** A teach pendant can be used to teach robot positions. This is a handheld programming and control unit. The ability to manually send the robot to a desired position, or "inch" or "jog" to adjust a position, are common features of such units. They also have the ability to change the speed, as a low speed is usually required for careful positioning or when testing a new or modified routine. In most cases, a large emergency stop button is also included. The teach pendant is typically no longer useful once the robot has been programmed.[15]

- **Lead-by-the-nose** Many robot manufacturers offer this technique. In this method, one user holds the manipulator of the robot while another enters a command that de-energizes the robot, causing it to go limp. The user then manually moves the robot to the required positions and/or along the required path, while the software records these positions in memory. The program can then direct the robot to these positions or along the predefined path.[15]

- **Robot simulation** These tools make it possible to write and debug robotics programs off-line, with the final version of the program tested on an actual robot. The ability to simulate a robotic system's behavior in a virtual world allows a variety of mechanisms, devices, configurations, and controllers to be tried and tested before being applied to a "real world" system. Robotics simulators can provide real-time computing of an industrial robot's simulated motion using both geometric modeling and kinematics modeling.[15] This will be used in the chapter 6.

The angles of each joint or displacements of the linear axes are the only parameters required for a given robot to completely locate the end effector (gripper, welding torch, etc). (or combinations of the two for robot formats such as SCARA). The most common and easiest way to define a point is to give it a Cartesian coordinate, which is the position of the 'end effector' in mm in the X, Y, and Z directions relative to the robot's origin. The robot controller must convert these coordinates to joint angles for a jointed arm, and such conversions are known as Cartesian Transformations, which may need to be performed iteratively or recursively for a multiple axis robot.[15]

Most articulated robots work by memorizing a set of positions and returning to them at various points in their programming sequence.[15] A simple 'pick-and-place' program for a robot moving items from one place to another can be in the following form (see figure 3.22):

*Define points P1–P5:*

1. Safely above workpiece (defined as P1)
2. 10 cm Above bin A (defined as P2)
3. At position to take part from bin A (defined as P3)
4. 10 cm Above bin B (defined as P4)
5. At position to take part from bin B. (defined as P5)

*Define program:*

1. Move to P1
2. Move to P2
3. Move to P3
4. Close gripper
5. Move to P2
6. Move to P4
7. Move to P5
8. Open gripper
9. Move to P4
10. Move to P1 and finish

**Figure 3.22:** A simple 'pick-and-place' program for a robot.[15]

### 3.3.3 Cobots or Collaborative robots:

Robots in the past were viewed as man-made devices that could easily injure or even kill a person due to his inattention. To avoid injury, one possible solution would be to remove human action from the robot workspace. The most common solution was robotic workplace safety fencing. However, in order for robots to be fully and productively engaged in some assembly areas, human intervention in the robot's workspace was sometimes required. This was one of the motivations behind the development of collaborative robotics, which allows and defines the sharing of a common space by a human and a robot.[68]

Collaborative robots' precise purpose is to produce or create something through common activities (cooperation) with humans. It is critical to emphasize that collaboration occurs in a collaborative workspace between a collaborative robot and a collaborative person.[68]

## Types of collaboration

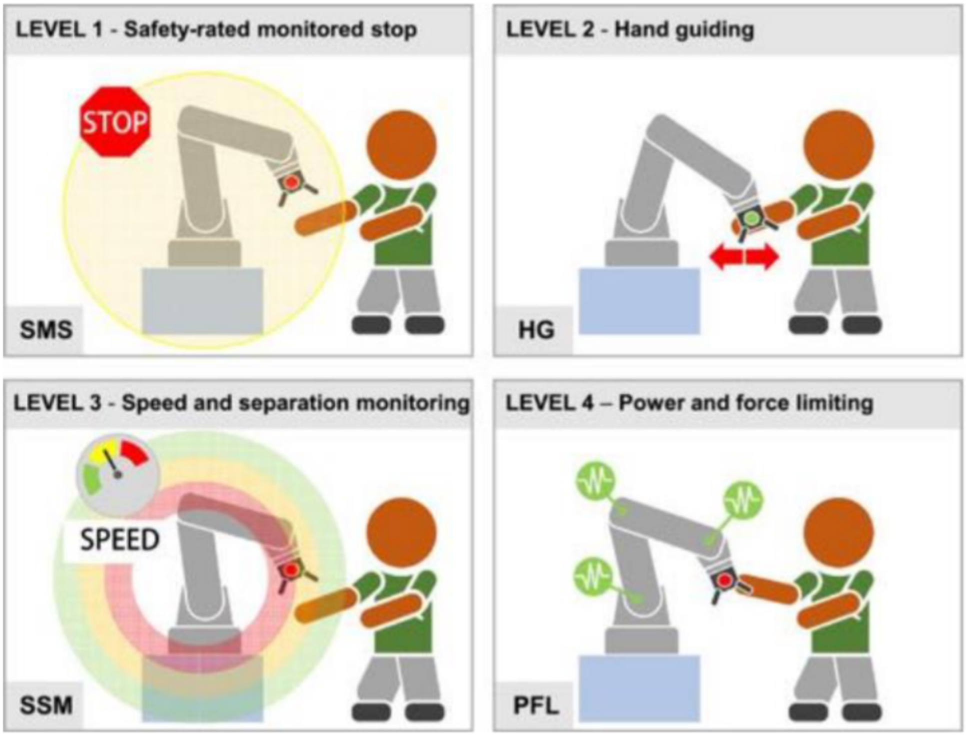


**Figure 3.23:** Types of human-robot collaboration.[68]

There are four types of human-robot workplace sharing:

1. **Safety-rated monitored stop:** This only allows a human and a robot to coexist. The robot is fenced, and when a human enters the defined area, all movement is halted.[68]

2. **Hand guiding:** This type is used to manually guide the robot along a trajectory that is memorized immediately.[68]

3. **Speed and separation monitoring:** The robot is no longer fenced in this mode, and the work area is divided into safety zones. The zones are monitored by laser scanners or a visual system, which slows the robot down and ultimately stops it altogether when a human enters the zone.[68]

4. **Power and force limiting:** the robots working mostly in this mode are called ***collaborative***. The robot is programmed to detect motor overload using sensors and to stop immediately when the allowed value is exceeded.[68]

The primary goals reported by manufacturing companies for introducing cobots into manufacturing processes are to improve productivity, flexibility, and quality. Managers and operators place a high value on other factors such as safety, ergonomics, trust, and potential future job trends. Interactions between robots and humans are intended to improve complex manufacturing processes, particularly when a robot can be guided by a worker and the robot provides power support to the worker, which is typically involved in difficult, monotonous, or exhausting tasks. Sensors, actuators, and data processing advancements have increased the level of assistance and support. Humans should only be involved in production processes when they are required; otherwise, they should focus on other tasks that will improve the overall system performance.[70]

**Figure 3.24:** A 6-axis Cobot from Hanwha manufacturer.[11]

Cobots or collaborative robots mostly have the same features or characteristics as the industrial robots (robots with 6 or more axes), so they can also replace the industrial robots in many applications. For more information on factors such as defined parameters, general benefits, end-of-arm-tooling, programming and simulation, the author also refers to these factors also mentioned in the previous subsection of Industrial robots3.3.2 as these factors are generally same for cobots in principle and function.

### Differences between Cobots and Industrial robots

Despite their similar functions, industrial robots and cobots have distinct features that distinguish them from one another, which is the role they play alongside humans.[4]

5 key differences between Cobots and Industrial robots:

1. **Employees are replaced by robots, while cobots work alongside them.**
   - Traditional industrial robots provide complete automation of their assembly line segment, allowing them to work without any human

assistance. They also frequently carry bulky equipment, such as large welding tools, while performing their duties with speed and dexterity.[4]

- Meanwhile, a cobot is frequently used to assist a human operator. It can carry out tasks that would be dangerous, exhausting, or time-consuming for an employee to carry out alone. It may also be involved in more complex tasks that cannot be fully automated, such as handling the wires within an appliance.[4]

2. **Robots must be kept behind cages or fences, which cobots rarely require.**

   - In terms of sheer speed, robots continue to outperform cobots, but they are designed with output volume over human safety in mind. Even if the robot comes into contact with an employee and injures them, the robot will only stop if the appropriate command is given, so the robots are kept behind fencing or cages.[4]
   - Human safety is a major concern in the design of cobots, which have built-in safety mechanisms that meet set safety requirements for human collaboration. If a cobot detects a human nearby, it may move more slowly, eventually stopping if the human gets close enough. If something gets in the way of its operation in the middle of it, it immediately stops and waits for a command. As a result, the cobot may not even require a safety cage.[4]

3. **Robot programming necessitates prior knowledge, whereas cobot programming does not.**

   - An employee must have advanced computer coding skills in order to give a robot instructions. Furthermore, most robot manufacturers have their own programming language that the employee must learn in order to do anything.[4]
   - An employee directing a cobot may not require any stock coding knowledge at all. Cobots have an easy-to-use interface that does not require any coding. Depending on the needs of the production chain, a single cobot can be reprogrammed and redeployed several times.[4]

4. **Robots remain stationary, whereas some cobots are mobile and adaptable.**

   - Robots are frequently bolted to the ground. Once they've started working, they usually don't need to be moved around the factory. This may also be regarded as a safety requirement, as it keeps the robot in place regardless of the speed and power required for its task.[4]
   - Cobots are typically light enough to be carried by a single worker. Employees can also easily move them around by using mobile bases.

The majority of cobots can also be mounted on walls or ceilings. There are also a variety of end effectors to choose from, such as grippers and drills, which adds to the versatility of cobots.[4]

5. **Robots are better suited to high-volume jobs, whereas cobots are better suited to complex tasks.**

    - Robots are still best for tasks with high quotas or hazardous conditions due to their speed and complete automation (e.g. exposure to lead or high temperatures).[4]

    - Cobots are easier to set up, more adaptable, and better suited to jobs that require precision.[4]

Cobots may be able to function without human assistance, but they are still best suited for jobs that require the cooperation of a human operator. By definition, neither a robot nor a cobot is superior to the other, as it is still subject to the demands of the assembly line.[4]

- Also, another great feature of cobots ,which comes from their lightweight nature and their easy compatibility with other components, is to combine them with external linear units (see subsection 3.3.1) to give the cobots an extra axis of movement, and can be installed as such in different orientation angles as per the linear 'slide', thus, increasing their already versatile use.[38]

### EOAT (end-of-arm-tooling) for Cobots

The EOAT becomes a critical component in making the most of cobots. These tools are intended to be safe around human workers and simple to program using the robot's teach pendant. They are cost-effective, highly adaptable, and simple to change for different processes, allowing for a quick ROI and results.[5] This also makes EOAT requirement an important factor while considering differences between cobots and industrial robots:

| TRADITIONAL ROBOTS | COBOTS | COBOT EOAT REQUIREMENTS |
|---|---|---|
| **Big batches, little variability**<br>Ideal for large companies that manufacture high volumes of the same products for long periods | **Low-volume, high-mix**<br>Designed for low-volume, high-mix production, where the robot is often redeployed for new processes | **Easy to change**<br>Flexible, quick-change tooling to eliminate downtime between processes |
| **Complex deployment**<br>Requires extensive programming skills and takes days or weeks to set up | **Fast and easy deployment**<br>Easy to deploy with simple programming that inexperienced users can set up in minutes | **Easy to program**<br>Tooling that is fast and easy to program and deploy using the cobot's built-in teach environment |
| **Requires constancy**<br>Programmed for unchanging environment and the same movement with minimal need to adapt | **Adapts to environment**<br>Flexible to adapt to changing environment and workpieces to be handled | **Adaptable**<br>Tools that easily adapt to varying sizes, shapes, and conditions of workplaces and the environment |
| **Not safe without guarding**<br>Typically requires safety guarding to keep human workers out of the robot's work cell | **Collaborative and safe**<br>After risk assessment, humans can work alongside robot in collaborative applications | **Design for safety**<br>Safe, collaboratively-designed tools that simplify interaction with humans |
| **Focus on the robot**<br>Repeats the same actions for years, with unchanging tool that is integrated for a specific process | **Focus on the EOAT**<br>As robot arm becomes a commodity, focus shifts to EOAT to increase robot utilization | **Flexible**<br>Flexible tooling that can be used for multiple processes |
| **Big investment, longer ROI**<br>Expensive robots, system integration, and operator training requires larger upfront investment and takes longer for ROI | **Lower upfront cost, faster ROI**<br>Competitive pricing, in-house integration, and ease-of-use minimize upfront costs and speed integration, uptime and ROI | **Faster integration**<br>Cost-effective tooling that speeds integration and reduces need for additional equipment |

**Figure 3.25:** Differences between Cobots and Industrial robots based on EOAT requirements.[5]

## Cobot grippers/EOAT available in market

In cobots industry, the EOAT is generally known as a 'cobot gripper'. These grippers even from different manufacturers (such as Robotiq, OnRobot, DH Robotics) are universally fit for all cobots available in the market today due to the presence of 'quick-changer' (figure 3.26) connected at the end-of-arm port. Also, there are quick-changers (figure 3.26) which can attach 2 or more cobot grippers/EOAT. This increases the usability prospects of the cobots as one cobot can do tasks which requires two different tools at the same time. Also, these grippers come in plug-and-play design, so it reduces their deployment time in real industrial applications as well as their programming time.

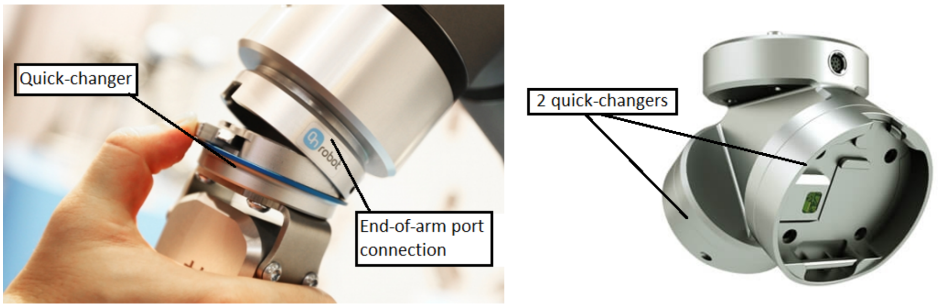


**Figure 3.26:** Quick-changer connected to end-of-arm port and dual-quick changer from OnRobot.[28, 8]

Here is a quick look over some of the cobot grippers available in the market suitable for different applications (some of these will be later used by the author in finding the suitable gripper for proposed solution in section 4.2 in the next chapter 4):-

1. **2 or 3 finger gripper:**

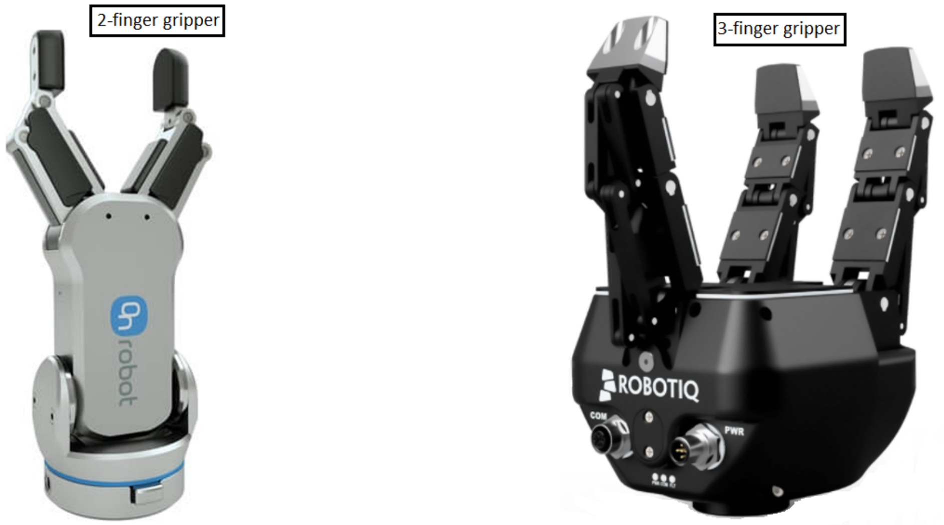


**Figure 3.27:** 2-finger gripper from OnRobot and 3-finger gripper from Robotiq.[34, 1]

This is one of the most commonly used type of cobot grippers in the industries. This type of grippers (figure 3.27) usually have very flexible design in the fingers (usually 2 or 3 configurable joints in each finger) to adapt to the objects shape and size for a solid grip which makes them very ideal for 'pick-and-place' tasks.

2. **Parallel stroke gripper:** This type of cobot gripper (figure 3.28) is very similar to the 2-finger gripper but differs in the finger design in

terms of finger flexibility. This type of gripers only uses parallel stroke to grab light and less complex objects compared to 2 or 3-finger gripper. This gripper is used in 'small parts industries' due to less complexity, less costs and configurable force and position control and also ideal in tight spaces[30].


**Figure 3.28:** Parallel stroke grippers available from DH Robotics.[7]

### 3. Electrical/pneumatic vacuum gripper

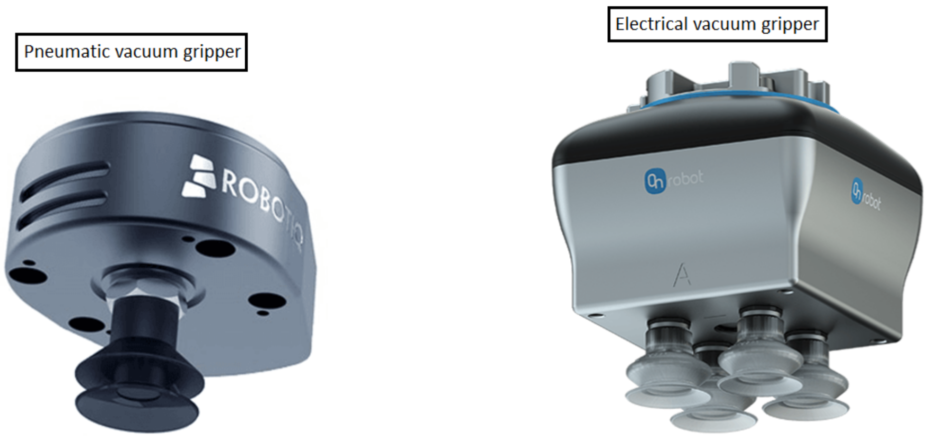


**Figure 3.29:** Pneumatic vacuum gripper from Robotiq and Electrical vacuum gripper from OnRobot.[49, 51]

This type of gripper uses vacuum suction pads to grab and lift objects by using - either compressed air through pneumatic vacuum pump or through electrical vacuum without external air supply. This type of gripper is ideal for objects of non-regular shapes especially in food industries and also works well in tight spaces. Another great feature of this type of gripper is the modular system of using vacuum suction pads, it's possible to combine multiple vacuum suction pads to lift bigger objects with large surface area such as carton packaging boxes, an example is shown in figure 3.30.

**Figure 3.30:** Electrical vacuum gripper from OnRobt with multiple vacuum suction pads.[50]

4. **Soft gripper:**

**Figure 3.31:** Soft grippers available from OnRobot.[41]

This soft gripper from OnRobot comes with interchangeable silicon-molded soft flexible cups to handle a wide array of irregular shapes and delicate items, making it ideal for use for 'pick-and-place' applications in food and beverages industries (due to food-grade material), cosmetics and pharmaceuticals industries as well as manufacturing and packaging industries.[41] This uses electrical vacuum to grab objects with no external air supply.

5. **Force/Torque Sensor:**


**Figure 3.32:** Hex Force/Torque sensor available from OnRobot.[14]

This 6 axis force/torque sensor offers 6 degrees of force and torque measurement, which makes this sensor ideal for direct contact-based applications like complex sanding, deburring, surface finishing tasks as it allows a precise control over the end-of-arm-tooling's applied force and ensures a constant speed and force throughout the contact-based application. This sensor deploys optical vision based technology also for path recording which is useful when it comes to most contact-based applications.[14]

### 3.3.4 *Karakuri* (Low Cost Automation):

*Karakuri* is a Japanese automation mechanism that was invented between the 17th and 19th centuries.[20]

For decades, *Karakuri* has been a key component of the 'lean manufacturing philosophy' and 'Toyota Production System,' where it refers to the simple but intelligent automation of processes based on physical principles – with no drives, sensors, electricity, or compressed air, just gravity and mechanical energy components like pulleys, counterweights and spring weight balancers (also known as square balancers).[20] One common example of *Karakuri* automation is ***flow racks*** used in factories to store and deliver parts or components to the operators on the production line (see figure 3.33).

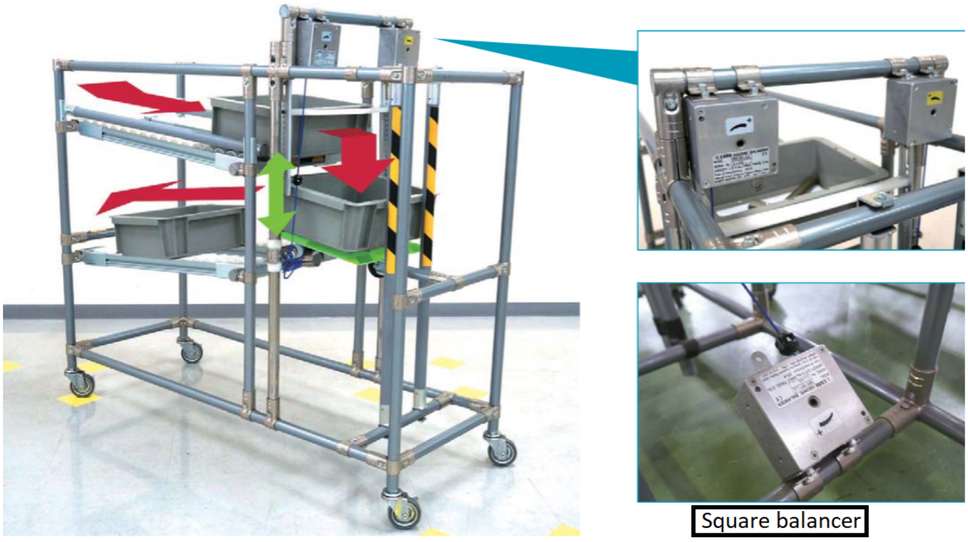


**Figure 3.33:** A real example of *flow racks* with the use of square (spring weight) balancers and pulley.[42]

Visualisation of the *Karakuri* automation of the aforementioned example is explained in the figure 3.34.

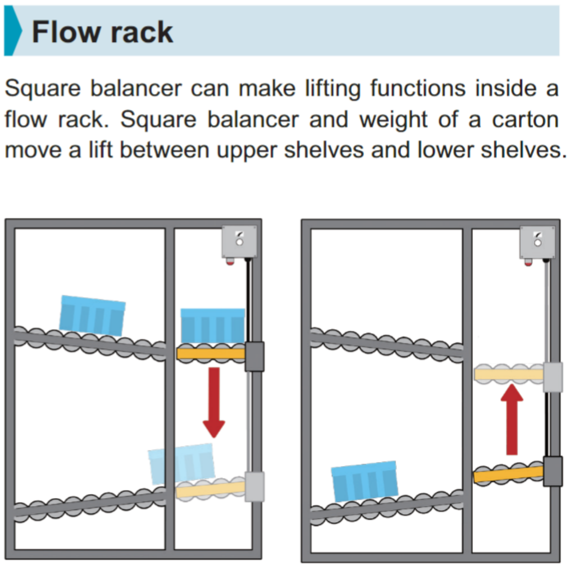


**Figure 3.34:** Explanation of self-delivering of boxes from upper shelves to lower shelves of *flow rack*.[42]

*Karakuri* or LCA (Low Cost Automation) typically costs 80 to 90 percent less than fully automating production processes while providing a comparable level of efficiency.[20]

### Hybrid *Karakuri*/LCA

The combination of muscle strength, leverage, and gravity is used in low-cost automation. Although this is applicable in mostly light lifting cases, it's not good to use *Karakuri* alone when it comes to heavy lifting cases or moving loads very high up. Then we have to use a combination of *Karakuri* and linear motion units and drives (see subsection 3.3.1) which forms the hybrid *Karakuri*. One example of this is already being used in TMMCZ Kolin (see figure 3.35) as a transport bridge for sending components' boxes to the workstation by lifting over the logistics routes. it uses electric linear units/drives to send the boxes up the ridge, then uses rollers to send to the other side and lowers the boxes by purely mechanical means.[21, 22]

**Figure 3.35:** Example of Hybrid *Karakuri* at TMMCZ.[21, 22]

# Chapter 4

# Search for optimal automation solution

After understanding the concepts and applications of various industrial automation technologies mentioned in the previous chapter 3 and based on the real-time observation of the degreasing process in TMMCZ Assembly line, the author has devised some factors necessary for proposing the automation solution for the given problem (see section 1.1):

1. **Complete automation of degreasing process:** the degreasing application needs to be fully automated to solve all the problems arising from the current application, meaning there is no need for collaboration between automation technology such as robots/cobots and human operators.

2. **Efficient use of space in the Assembly line:** as per author's discussion with the Process design team at TMMCZ for the changed Assembly line layout (see figure 4.1), there is fully available only one side of the Assembly to install all the automated technical equipment. This proposed installation area (area marked in purple in figure 4.1)is before 'Pitch 0' while the current degreasing process takes place at 'Pitch 2' and glass installation at 'Pitch 3' (see figure 1.4).

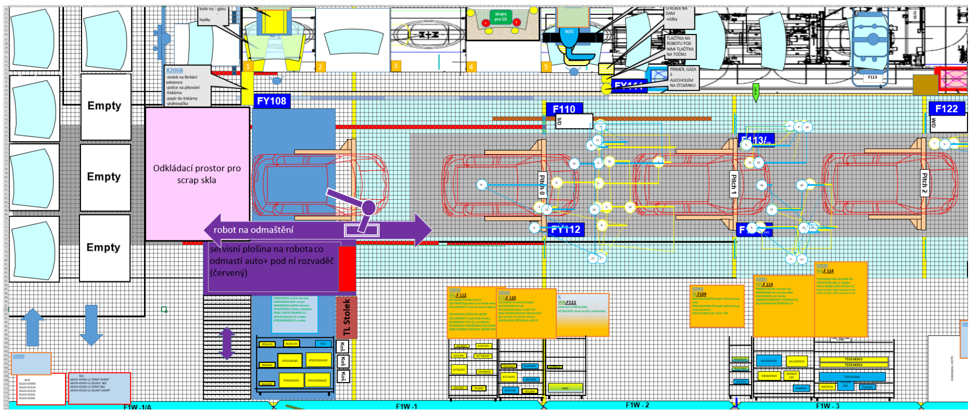


**Figure 4.1:** Modified layout of TMMCZ Assembly line for degreasing workplace.[62]

3. **Need for automation equipment to be adaptive, quick and flexible:** as the Assembly line with the car body carriers is continuously in motion, the automated technical equipment must be quick and adaptive enough to be able to accurately move along the car body carrier in the selected travel path and stop when the Assembly line stops. At the same time, it should be flexible enough to work for different car models.

4. **Need to maintain no collision:** the continuously moving automation equipment should always maintain safe distance to avoid any collision with the car body or car body carrier during the motion and wiping application.

5. **Equipment should be compact:** as the equipment installation zone is in air higher than the base floor (see figures 4.1, 1.4 and 1.2), the equipment should be compact enough in size for installation away from the floor, with support from structures hanging from upper structure framework in TMMCZ factory.

6. **Equipment is preferred to be lightweight:** as discussed in the previous point and also, given that degreasing doesn't require any heavy payload or end-effector (EOAT), the equipment should preferably be lightweight.

7. **Safety of operators:** though the equipment is proposed to be installed in a zone further from the human operators, there must still be a safety system for the equipment to stop if any human interference occurs in order to avoid any injury or safety risks. For this, the proposed placement of automation equipments can install fencing on the sides of the Assembly line and a laser security scanner on the base floor to avoid any accidents or human interference.

8. **Soft material for end-effector tool (EOAT):** as the end-effector tool is directly in contact with the surface of the car body flange, it should have soft materials around so as not to damage the car body flange or the surrounding area in any way (as seen in figure 1.5, we can see that the width of application area on the body flange is not same, so this requires soft materials on the EOAT during the wiping application.). Also, it's ideal to use force-sensor (see figure 3.32) at the end of the EOAT to properly do the wiping application with adequate force to not leave any marks on the body flange surface.

## 4.1 Exploring different ideas for automation equipment

Based on the aforementioned factors and study of the automation technologies (see section 3.3), the author designs some possible automation setups to automate the degreasing process, primarily focusing on the equipment which will move along the car body carrier on the Assembly line and the type of technology (4-axis arm, robot, cobot) used to hold the application EOAT. *All the designs have been made by the author by using the design software* ***AutoDesk Fusion 360****[10] (the actual design files (.STEP format) can be referred to in the electronic attachments 8.1 and also in the embedded links in the figure descriptions (only in electronic version)).*

### 4.1.1 Initial idea

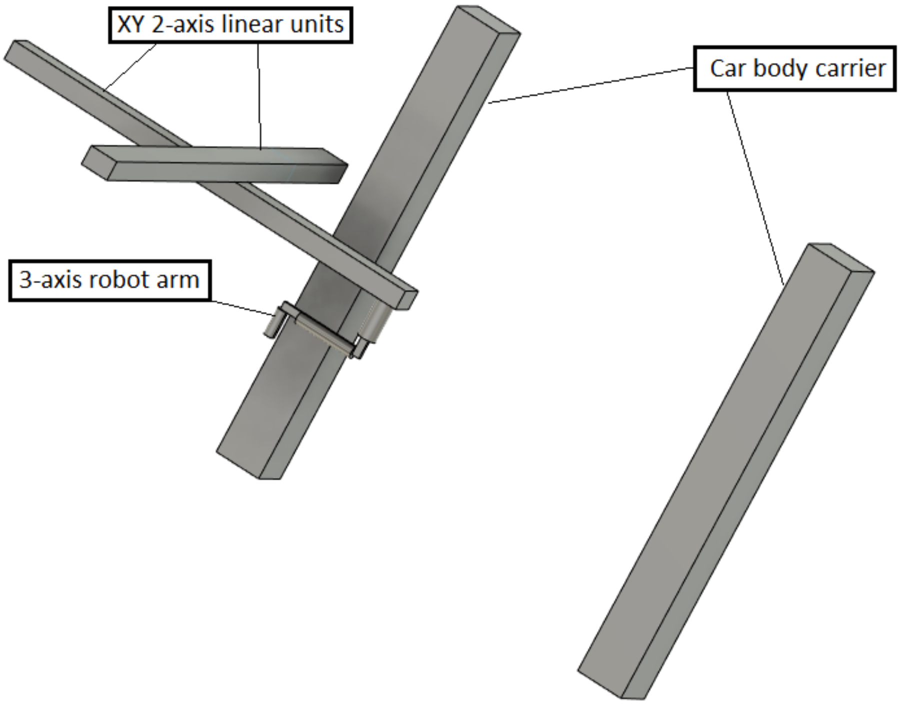


**Figure 4.2:** 1st idea

Inspired by the study of linear units (subsection 3.3.1) and past experience with linear technology in his engineering work at TMMCZ, the author proposed using a setup (figure 4.2) of 2 linear units in X-Y axis with the second axis holding a 3-axis robot arm to hold the end-effector tool (EOAT). The idea here is that the linear in one axis will follow the car body carrier in motion and the second linear will move the hanging robot arm to the degreasing application zone on the car body (windshield and back window area).

**Problem:** as the car body flange isn't a flat surface (see figure 1.5), the 3-axis robot isn't effective in wiping or contact based application with the surface of body flange.

### 4.1.2 2nd idea

The author tries a different approach by using a setup of a single-axis linear unit and a 6 or 7-axis robot arm attached on the side profile of the linear unit (see figure 4.3). The idea here is that the whole setup (including both linear unit and robot arm) as a whole will follow the car body carrier in motion, the robot arm making use of 6 or 7-axis will contract itself during motion and extend itself during the application on the car body flange.

**Problem:** given the size of the car body, the robot arm required for this idea has to be very long, which makes a huge risk of collision with the side pillars of car body carrier during motion while contraction and extension of its arm.

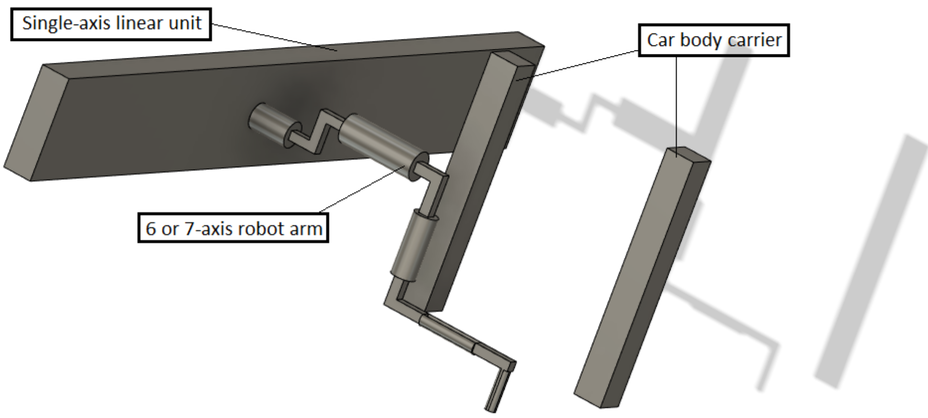


**Figure 4.3:** 2nd idea

### 4.1.3 3rd idea

The author tries a new idea by using single-axis linear unit and long arm cobot (due to its lightweight nature mentioned in 3.3.3) by hanging the cobot vertically on the linear unit. (Also, the author managed to get actual 3D models for the linear unit[18] and cobot[11]). There are three approaches in design which arise from this idea:

1. **1st approach**

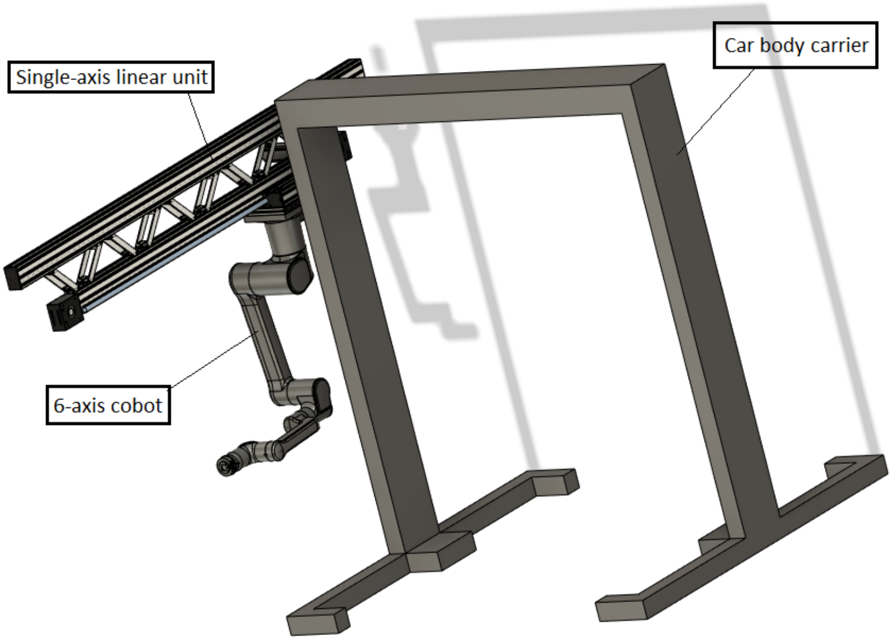


**Figure 4.4:** Single-axis linear unit with cobot hanging down

The approach here (figure 4.4) is to hang the long arm cobot down the single-axis linear unit, the 'slide' part of linear unit (see figure 3.8) will follow the car body carrier and the robot will extend itself and do the wiping application.

**Problem:** Although the flexibility and lightweight nature of cobots makes this approach a seemingly feasible option but at the same time may lead to a possible collision with the side pillars of the car body carrier during the cobot arm movement.

2. **2nd approach**

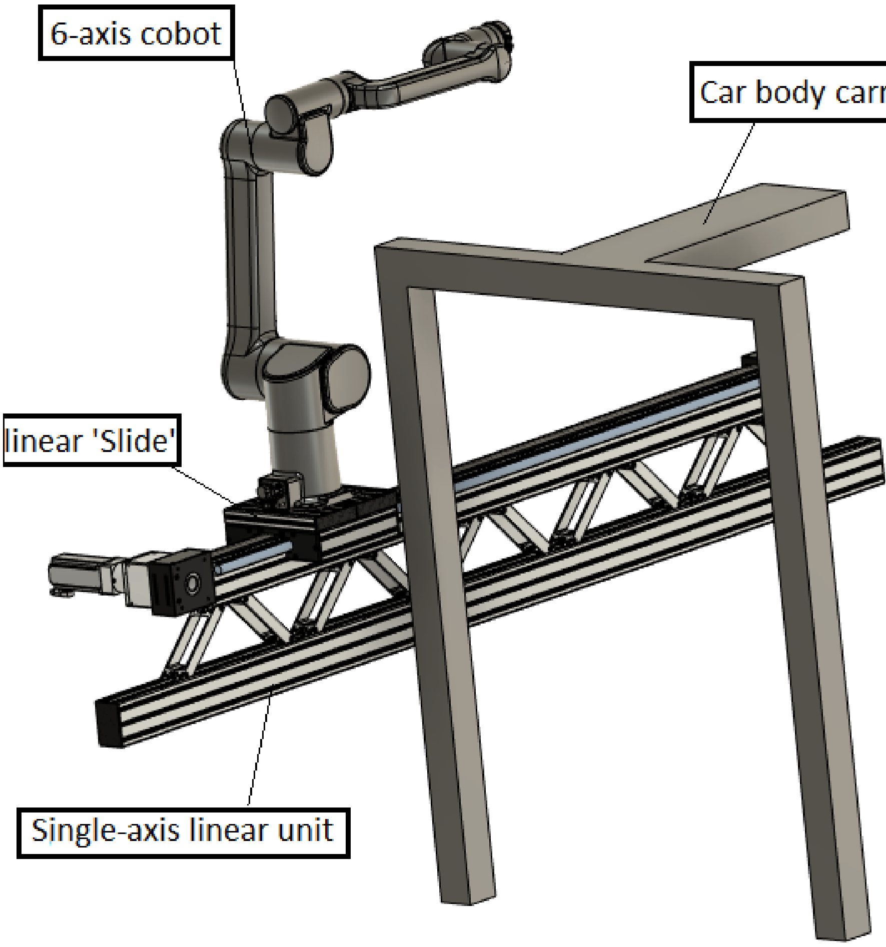


**Figure 4.5:** Single-axis linear unit with cobot standing above

The approach here (figure 4.5) is to dock the long arm cobot on top of the single-axis linear unit, rest of functioning is same as mentioned in the previous approach with the difference being only in the orientation angle of installation (upside down).

**Problem:** as the application area on car body flange is towards the top of car body, the cobot will have very congested space in the zone where it contracts and extends with respect to the car body carrier which will lead to possible collision with the car body carrier's side pillars.

3. **3rd approach**

Another possible approach (figure 4.6) is to use the 1st approach (figure 4.4) but on both sides of the car body carrier, thereby increasing the application speed and reducing process time. The 'slide' on the linear units on both sides will follow the car body carrier in motion, the cobots from both sides will extend themselves and do the wiping application by dividing the wiping perimeter into half each for themselves.

**Problem:** as per the new Assembly layout (see figure 4.1), there is only one side of the car body carrier which is fully available for installation, so the existing equipment and processes on the other side can interfere with the second setup (figure 4.6) of linear unit and cobot used in this approach.

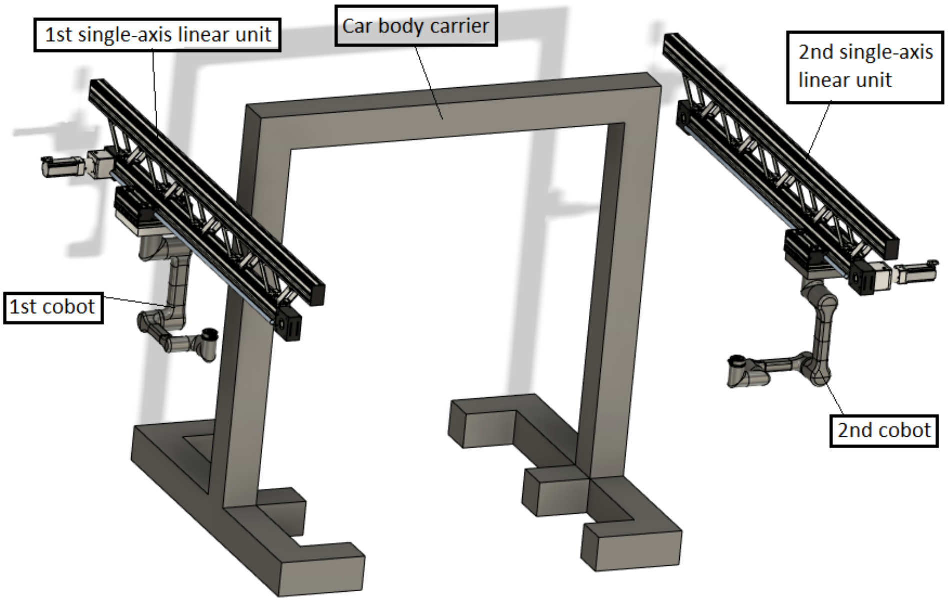


**Figure 4.6:** 2-sided single-axis linear units with cobots

After reviewing the three approaches to the idea of using single-axis linear unit and long arm cobot, the author concludes that 1st and 2nd approaches show potential risks of collision with the side pillars of the car body carrier and the 3rd approach is just not feasible enough given the limitation of available space for installation on the Assembly line workplace (see figure 4.1).

### 4.1.4 Final idea

After reviewing all the previous ideas, the author tries a new design idea (figure 4.7) incorporating features from all these previous design ideas. (Also, now the author has access to the actual 3D model of the real car body carrier to use for the design idea).

To solve the potential problems in previous design ideas such as efficient use of space, possible collision with the side pillars of car body carrier, congestion of space in application zone, the author proposes the idea (figure 4.7) to use a combination of single-axis linear units in XY 2-axis setup with 2 cobots hanging from the 2nd axis linear unit, the 1st linear unit's 'slide' follows the car body carrier, the 2nd linear unit moves back and forth carrying the 2 medium length arm cobots over the wiping application area.

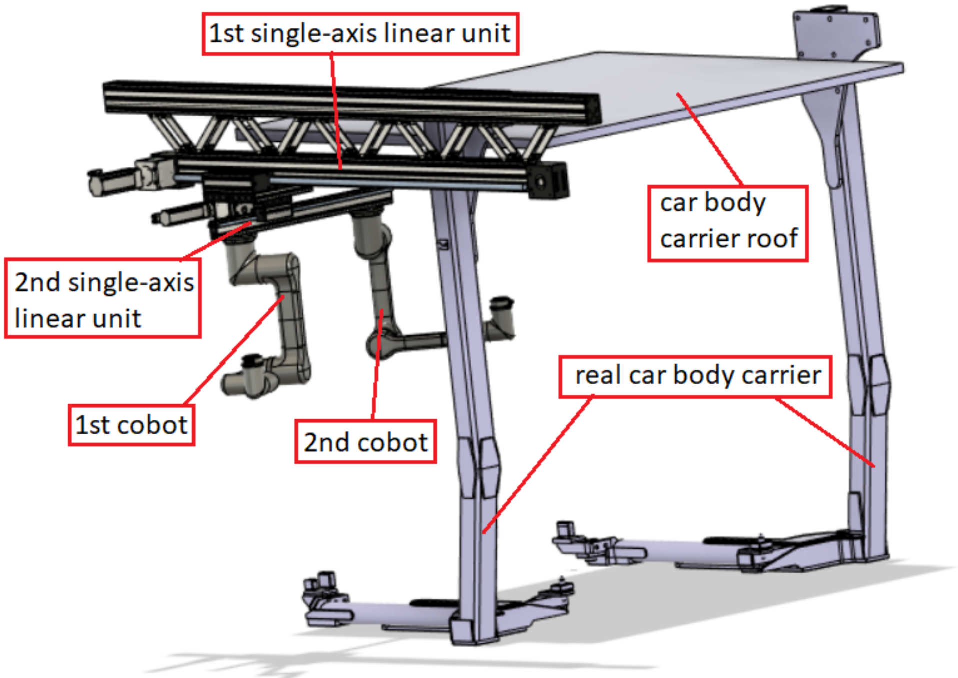


**Figure 4.7:** 2-axis linear units and 2 cobots with real car body carrier

The author will primarily use this design idea (figure 4.7) for simulation in chapter 6 to determine its overall feasibility in implementation.

## 4.2 Finding the optimal end-effector tool for wiping application

The next automation technology to configure after the linear units and cobots is the end-effector tool or end-of-arm-tooling (EOAT) for the cobots. The author makes use of the capability of cobots to have 2 or more tools on it's arm's end (see subsection 3.3.3).

The author designs three approaches of the possible gripper to do the wiping application:-

### 4.2.1 Alcohol spraying and air blowing:

The idea here (see figure 4.8) is to use two end-effector tools - one nozzle for alcohol/liquid spraying on the surface of car body flange and second nozzle for blowing compressed air to help clean the surface.

**Problem:** the author tried to test this idea using an alcohol spray and a compressed air gun to clean/degrease the surface of the car body flange.

While this method did clean the usual dirt and dust particles, but it couldn't clean the oil and grease marks, fingerprints and residual materials. This proves that the cleaning/degreasing method has to be direct contact-based with respect to the surface of the car body flange, by using cloth or wipe as in current manual degreasing (see subsection 1.1.2).

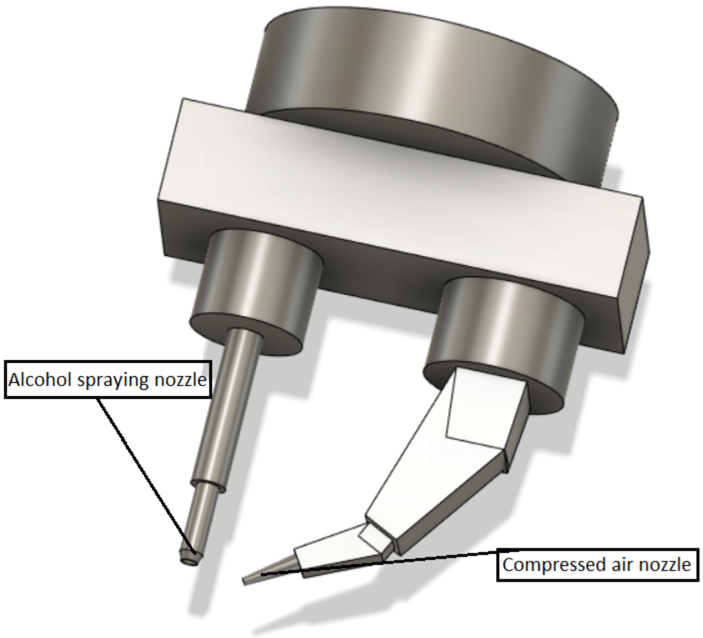


**Figure 4.8:** Cobot gripper with nozzles for liquid alcohol and comp. air.

### 4.2.2 Alcohol dispensing on a cloth piece

The idea here (see figure 4.9) is to combine a 2-finger gripper (see figure 3.27) with a liquid alcohol dispensing nozzle, the 2 fingers will pick up and hold the cloth piece and the nozzle will drop some amount of liquid alcohol on the cloth piece and then the cobot while holding the cloth piece (with 2 fingers) will do the wiping application on the surface of the car body flange.

**Problem:** one clear problem here is the over-complexity of the designed gripper which also increases process time with so many steps (dispense alcohol, pick up cloth piece, hold the cloth piece, do the wiping application, dispose off the cloth piece). Second problem is the highly inflammable and dangerous nature of 99 percent concentrated liquid IPA in such an automated process, bringing out huge potential of fire risks as well as requiring many extra safety certifications such as CE[44]. With this observation, the author prefers to go with pre-saturated wet wipes (subsection 1.1.2) considering their low risk factors and less process as the wipes are already wetted with liquid IPA directly from their manufacturer.

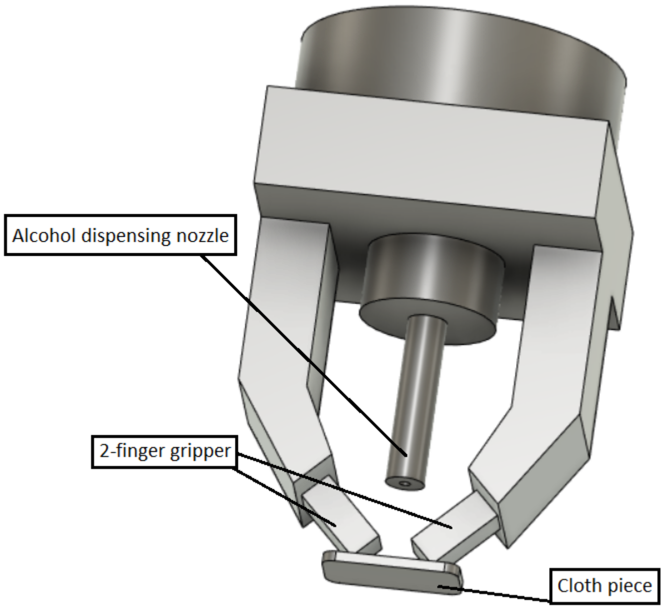


**Figure 4.9:** 2-finger gripper with nozzle for liq. alcohol

### 4.2.3 Finding cobot gripper suitable for use with pre-saturated IPA wet wipes

The author, already having tried and failed with soft gripper (see figure 3.31) and vacuum gripper (see figure 3.29) which were unable to pick up the wipe properly due to its much thinner form factor (1-1.5mm), the author decides to test parallel-stroke 2 finger gripper (see figure 3.28) with some modifications:

#### 1st idea

The idea here (see figure 4.10) is to cover the fingers of the parallel gripper (3D model from DH Robotics[7]) with foam-based or soft material and attach or bolt in foam-based block in the mid-section of the base of the gripper, this middle block will act as a base to hold the wipe while the 2 fingers will pick it up and do the wiping application.

To actually test this idea as this gripper modification isn't very complex, the author collaborates with one of the automation technology partners of **TMMCZ** called **HennLich s.r.o.**[36] to test their available parallel gripper with designed foam-based block in the mid-section of the gripper's base. (Gripper test video recording available in the electronic attachments 8.2.)

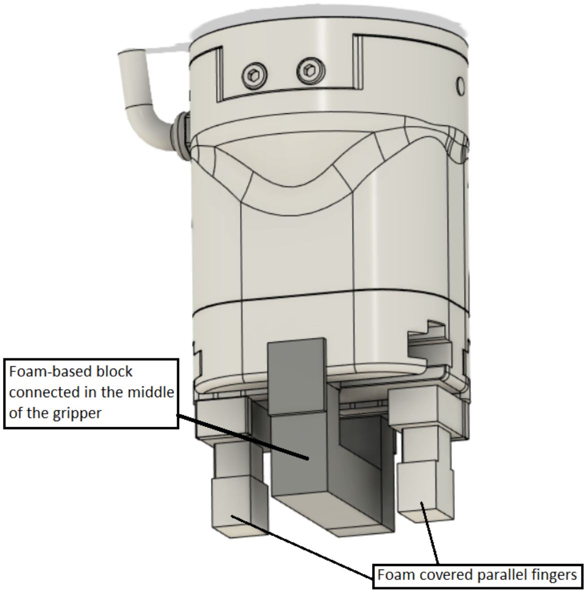


**Figure 4.10:** DH Robotics parallel gripper 3D model with foam middle piece

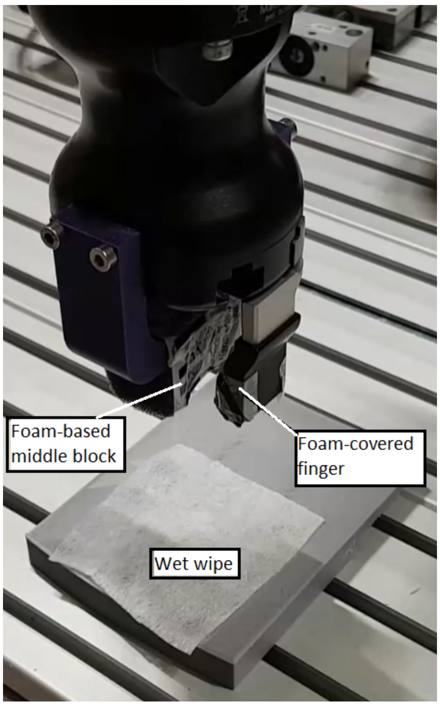


**Figure 4.11:** Actual test of parallel gripper with foam-based mid-section attachment at HennLich s.r.o. using Hanwha cobot.[36, 11]

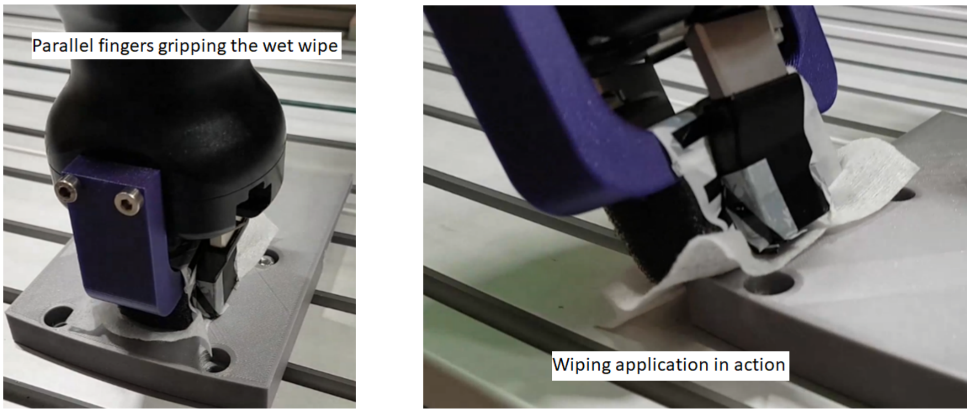


**Figure 4.12:** Wipe gripping and wiping application by the modified parallel gripper in action.[36, 11]

As seen in the figure 4.12 (and the embedded video in the caption), the gripper was able to grip the wipe and wipe the test-piece (metal slab) very well in the actual test.

**Problems:** one apparent problem is the bulky size of the gripper, which isn't feasible given the narrow cleaning surface area of the car body flange (see figure 1.5). Second problem is the lack of good grip of the wipe by the gripper as in the real test, the wipe could easily be taken off the gripper by a very slight force, thus, reducing its overall feasibility.

## 2nd idea

The idea here (see figure 4.13) is to attach a foam-based block on each finger of the parallel gripper, the fingers will have full parallel stroke to fully hold the wipe after picking it up with a good grip.

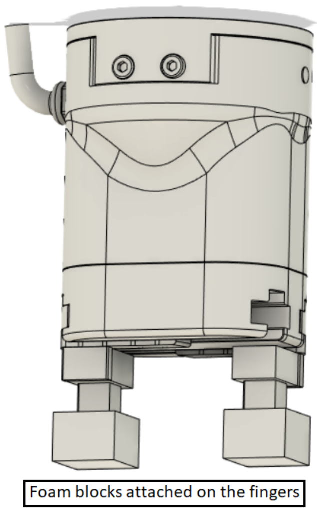


**Figure 4.13:** DH Robotics parallel gripper with foam blocks on fingers

As this modification isn't complex to test, the author again collaborates with **TMMCZ's** automation technology partner **HennLich s.r.o.**[36] to test their available parallel gripper with fingers attached with foam blocks to fully grip the wipe. (Gripper test video recording available in the attachments 8.2.)


**Figure 4.14:** Actual test of parallel gripper with foam blocks attached to the fingers at HennLich s.r.o. using Hanwha cobot.[36, 11]


**Figure 4.15:** Testing of gripping of wet wipe and strength of grip using foam blocks.[36, 11]

As seen in the figure 4.15 (and embedded video in the caption), the gripper was able to grip the wipe with a strong hold in the actual test.

**Problems:** one apparent problem is again the size of the foam blocks although being a soft material, it won't cause any damage to the car body or its carrier but will be a difficulty in proper wiping application. Second problem is the surface area of the wipe to be used after being gripped by the foam blocks, the actual car body flange has some much smaller areas for wiping or degreasing (see figure 1.5.

### 4.2.4 Final design of gripper

As per the real testing results, the parallel gripper proves to be the best gripper base choice but will need some extra modifications to make it ideal for use on smaller surface areas on the actual car body flange. The author will design this in the next chapter 5 after designing an automated solution for the wet wipes dispensing machine, this is necessary as the design of the wet wipes dispensing machine and the cobot gripper are interdependent due to the wet wipes being the common object/material to use.

## 4.3 Finding automated system to dispense the wet wipes

The pre-saturated wet wipes come in form of rolls of 100-500 wipes each all packed in plastic canisters to not dry (see figure 4.16), depending on the manufacturer.

**Figure 4.16:** IPA pre-saturated wet wipes in a roll and plastic canister.[19]

For the usage, the wipe is manually torn from the roll, so this process will need to be automated before making a system to dispense the wipes. For this, the author tried to use a system similar to the ones used in vinyl wrap cutting machines (see figure 4.17) which uses a moving cutting blade to cut the vinyl wraps, the author was able to test this on a plastic wrap cutting machine (used for car body) at **TMMCZ** but the normal cutting blades couldn't cut the *'lint-free non-woven'* wet wipes properly.


**Figure 4.17:** Vinyl wrap cutting machine with the moving cutting blade.[55]

## 4.3.1 Ultrasonic cutting

Now, the author tries **'ultrasonic cutting system'**, which uses vibrational frequency for its cutting blade to be able to smoothly cut difficult-to-cut materials. The ultrasonic cutter is a safe and clean processing machine that emits no cutting chips, polluted water, noise, or smoke. The cutting blade expands and contracts at an ultra-fast rate of 20,000 times or more per second, with an amplitude of up to 70 microns. As a result, it can easily and beautifully cut materials that are normally difficult to cut.[47]

The ultrasonic cutter (see figure 4.18) is made up of a "transducer," which produces vibration, and a "oscillator," which drives the transducer. The transducer is made of a piezoelectric element. When a voltage is applied, the piezoelectric element causes the transducer to move a few micrometers. Vibration is produced when a voltage is applied on a regular basis. Each object has a unique frequency that allows it to be stable and easy to vibrate. A small force can produce a large vibration by adding an external force that corresponds to that special frequency. This is referred to as resonance. The piezoelectric element in the ultrasonic cutter generates a force that resonates the entire body, from the transducer to the blade tip, resulting in a large vibration at the tip. The oscillator generates voltage on a regular basis in order to resonate and drive the transducer. A larger vibration can be obtained by wringing the cross-sectional area from the piezoelectric element to the blade tip using a component of the ultrasonic cutter called the horn.[47]


**Figure 4.18:** Oscillator and transducer with cutting blade from SonoTec manufacturer.[47]

As the ultrasonic cutter is ideal for fast, precise cutting of difficult-to-cut materials (such as fabric, rubber), it is used in a variety of industrial applications in connection to linear units or 3-axis robots (see figure 4.19) or even cobots. The author decides to use this system to cut the wipes in the automatic wipes dispensing machine which will be designed next.

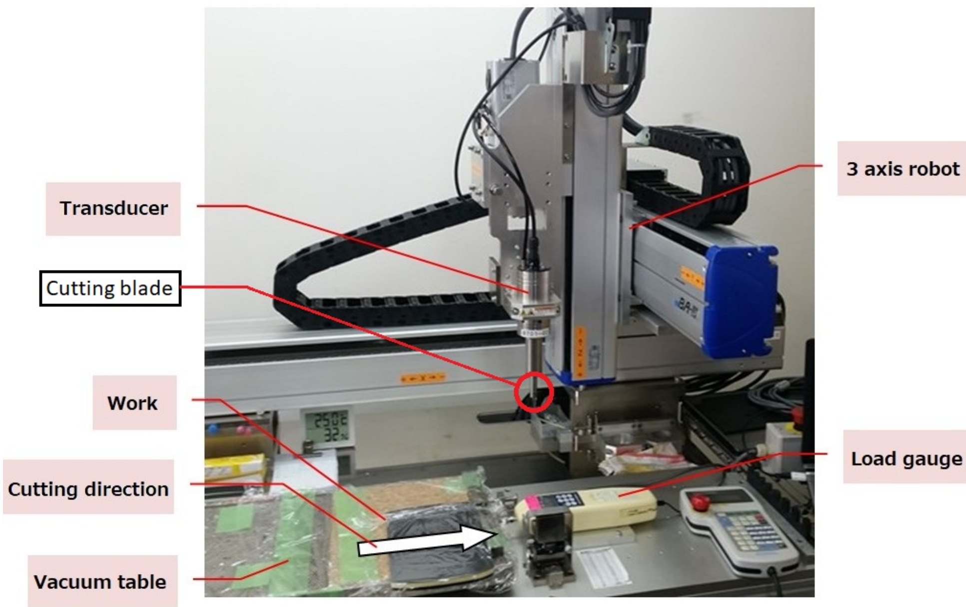


**Figure 4.19:** Ultrasonic cutter being used with a 3-axis robot.[48]

To test this linear moving ultrasonic cutting on the wet wipes, the author collaborated with an ultrasonic provider company called **DJK Europe**[56] to test using their setup of a linear manipulator with an ultrasonic cutter (see figure 4.20). This test required one human operator to hold the wipe from the other side to not let the wipe get loose while being cut, this can later be automated by using vacuum pad grippers (see figure 4.23). The cutting results (see figure 4.21) were excellent as the ultrasonic cutting blade cut the *'non-woven lint-free'* material of wet wipes very efficiently with precision and high speed. (Ultrasonic cutting test video recording available in the electronic attachments 8.2.)

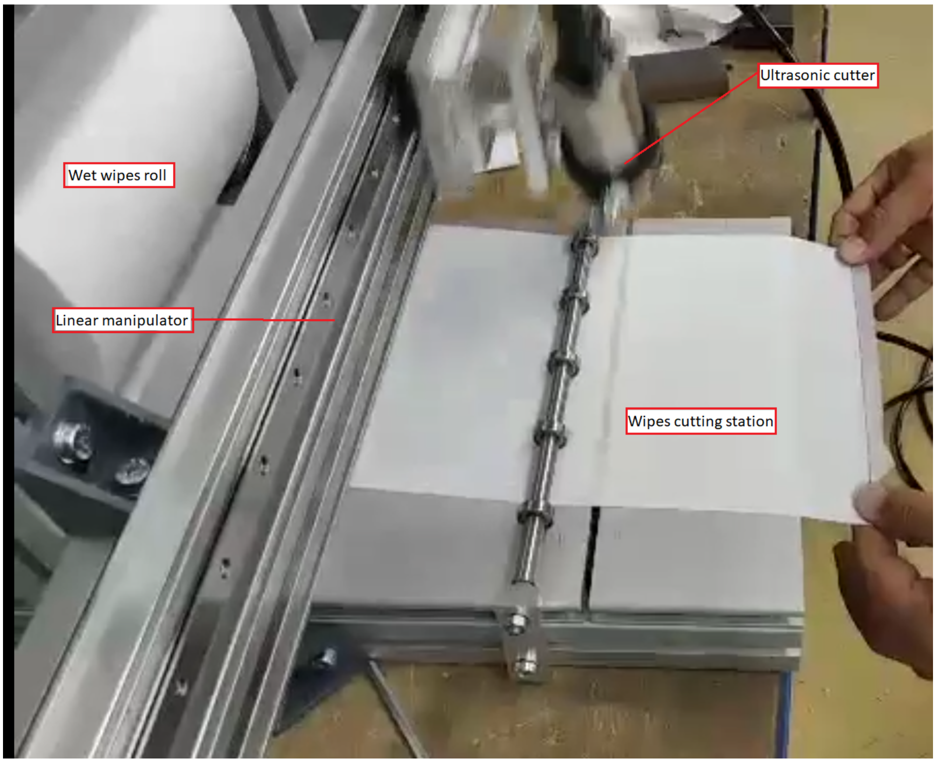


**Figure 4.20:** Test of ultrasonic cutting of wet wipes at **DJK Europe**.[56]

**Figure 4.21:** Wet wipe was cut precisely at a high speed.[56]

- The automated wipes dispensing machine will be a single-purpose machine as there is no available automated solution for cutting and dispensing of wipes, so the author will incorporate different technological solutions together to design one machine for wipes cutting and dispensing.

### 4.3.2 First design idea

The author's idea here (see figure 4.22) is to store the wet wipes roll in an airtight chamber or compartment with a door on one side so that the roll can be replaced with a new one when fully used. From here, the wipes will be pulled forward by vacuum pad grippers (see figure 4.23), at the correct position, the ultrasonic cutter will move in a linear direction with a linear manipulator (see figure 3.16) and cut the wipe which will then be picked up by the robot from the last platform on the wipes machine. The author proposes to use PLC controls to operate this machine and to connect with the main linear unit-cobot system and gripper.

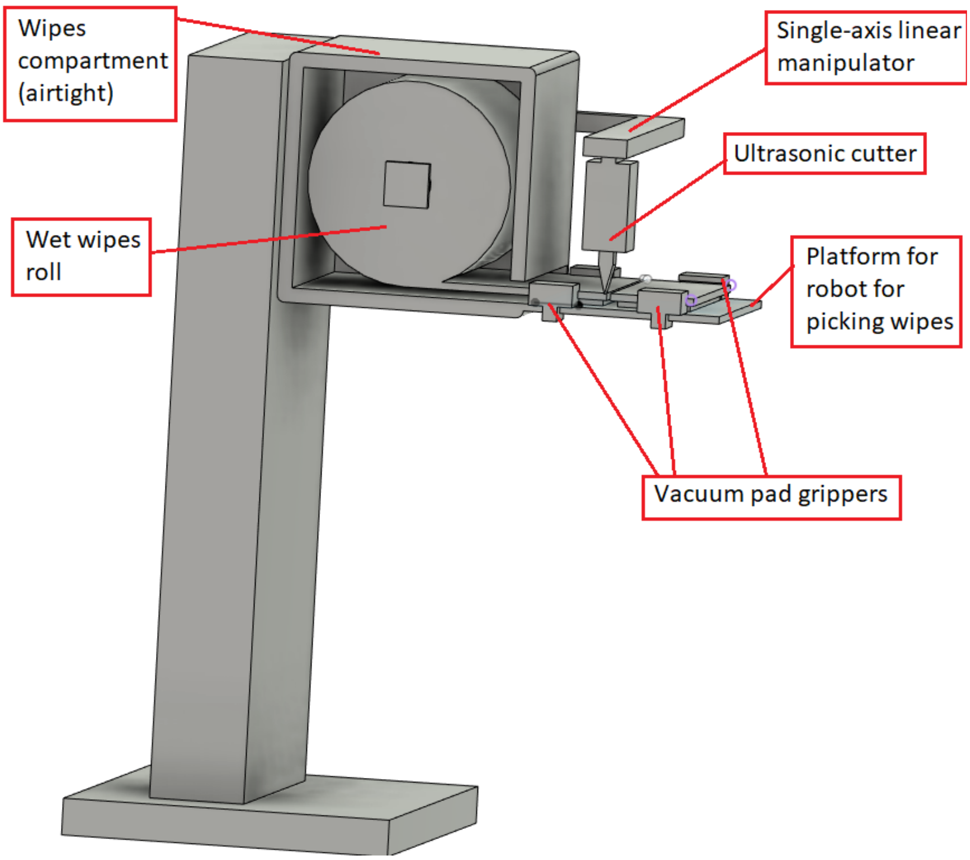


**Figure 4.22:** First design idea for wipes cutting/dispensing machine

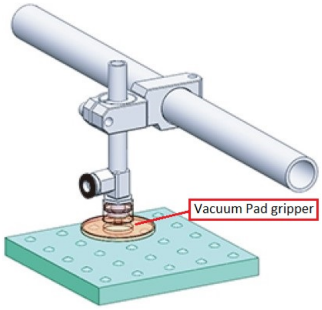


**Figure 4.23:** Vacuum pad gripper.[9]

## 4.4 Optimal solution

After comparing and testing of different design ideas, the author decides to continue with the chosen linear unit-cobot system (see figure 4.7), the chosen base gripper (see figure 4.13) which will be modified in the next chapter 5 to fix its shortcomings and the chosen automated wipes cutting/dispensing machine (see figure 4.22).

# Chapter 5

# Complete design of automated solution

As discussed in the section 4.4, there are three main components of the whole automated car body degreasing system - wet wipes cutting/dispensing machine, linear unit-cobot system and cobot gripper. Continuing with the initial designs from the previous chapter 4, the author will proceed to make final designs of these three main components and then the overall automated degreasing process design.

## 5.1 Automated wet wipes cutting and dispensing machine

First redesign is for the airtight chamber/compartment for storing the wet wipes roll with a door to replace the roll after full use.

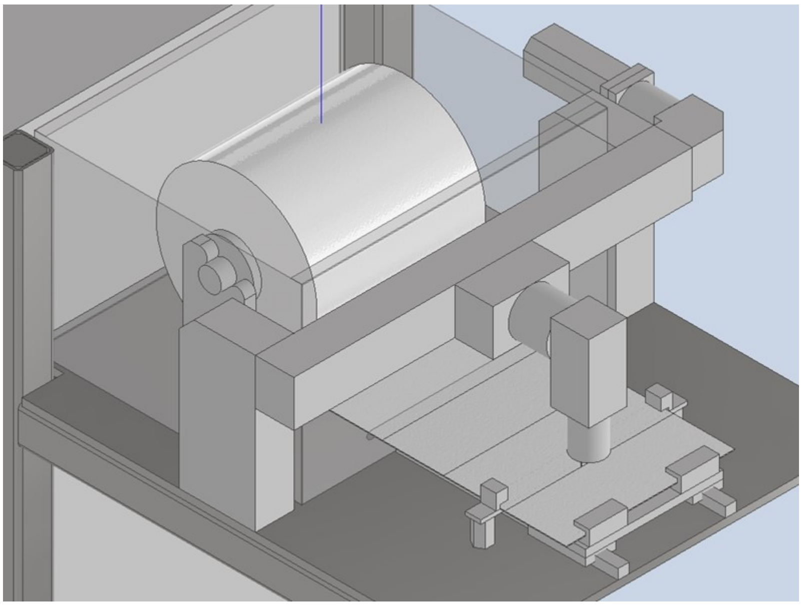

**Figure 5.1:** Inside view of wet wipes roll compartment

The author proposes to make the top cover as the door which can be opened to replace the roll. Rest of the redesign (see figure 5.2) remains mostly similar to the base design (see figure 4.22).

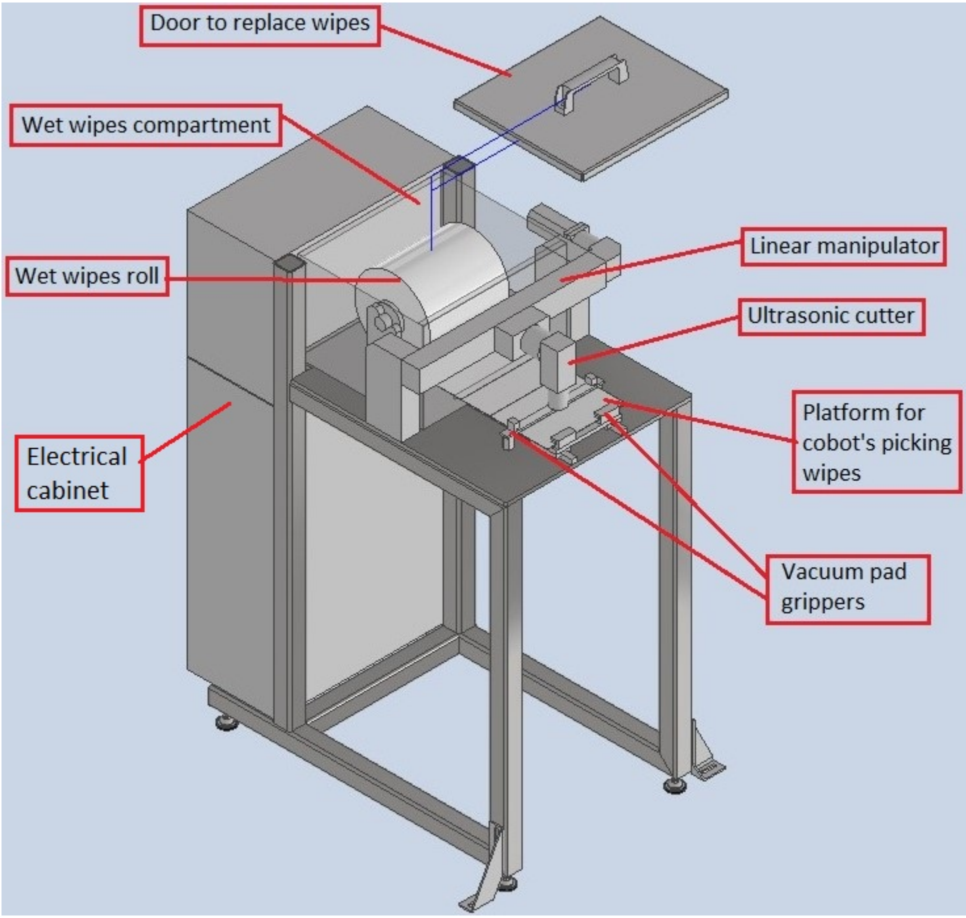


**Figure 5.2:** Final design for wipes cutting/dispensing machine

## 5.1.1 Wipes pickup platform for cobot

The author redesigns the platform (see figure 5.3) where the cobot will pick up the cut wet wipe from. The idea here is to make a gap in the middle of the platform, so the gripper picks the wipe in such a way that the mid-section of the wipe is the end-point of the gripper (to be designed in the next section 5.2), the side parts of the wipe will be held by the parallel fingers of the gripper. There are separate vacuum pad grippers, one pair for pulling forward the wipe from the wet wipes roll and second pair to move the wipe after cut to the cobot pickup platform. This design idea will be more understandable after the design of the modification of the parallel gripper in the next section 5.2.

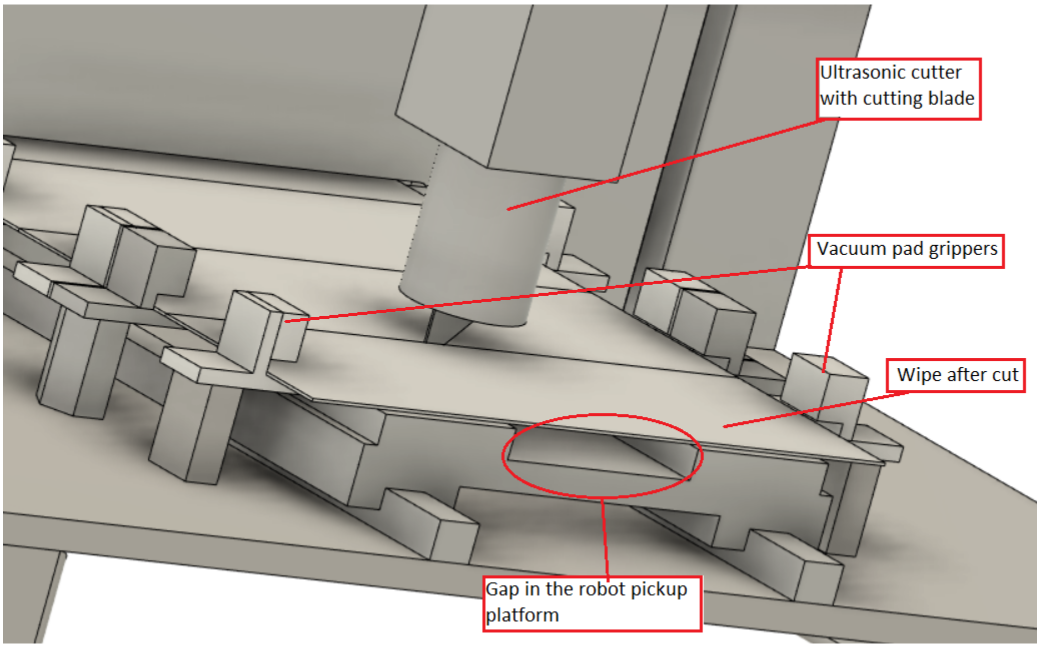


**Figure 5.3:** Final design for wipes cutting/dispensing machine

## 5.2 Final modification of parallel gripper

As discussed in the previous subsection 5.1.1, the author makes a modification (see figure 5.4) in the previous parallel gripper design (see figure 4.13).

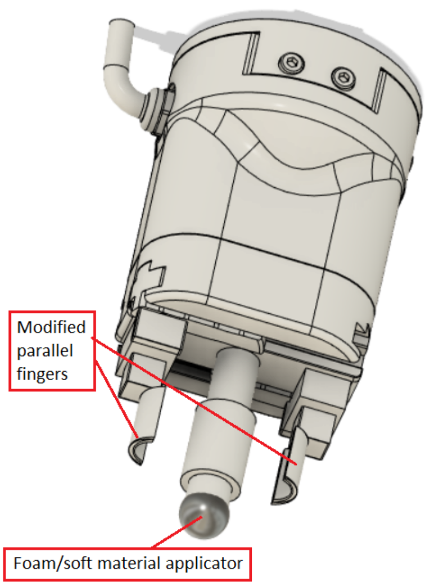


**Figure 5.4:** Final design for modified cobot gripper

As discussed in the subsection 5.1.1, this modified gripper will be suitable for pushing the wipe in the gap of the platform while the parallel fingers will grip the wipe and then this soft middle extruded part (can be 3D printed) of the gripper will act as an applicator for wiping the car body flange as it's small enough to wipe the narrow areas of the body flange as well.

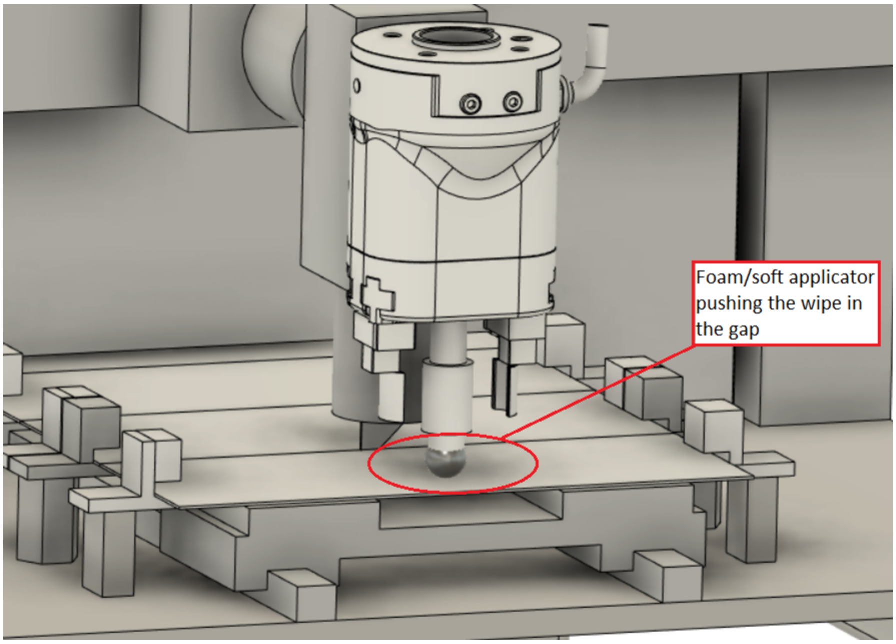


**Figure 5.5:** Modified gripper on the wipe pick up platform of the wipes cutting/dispensing machine.

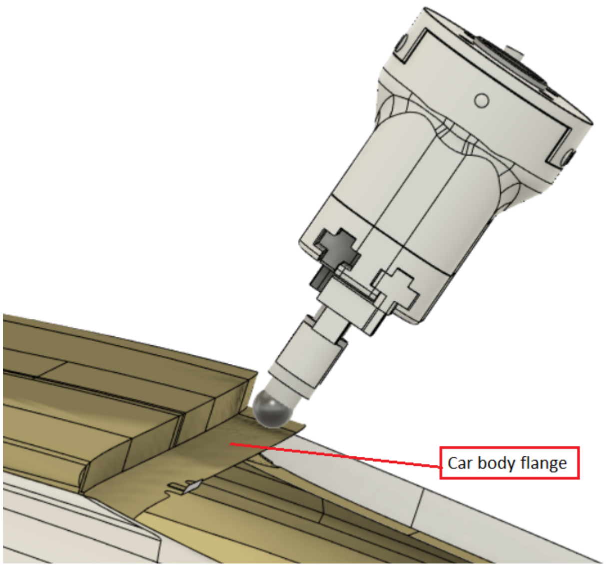


**Figure 5.6:** Testing modified gripper's applicator on the narrow area of car body flange.

## 5.3 Linear unit-cobot system

Continuing with the chosen design (see figure 4.7) and access to 3D model of real car, the author tests this design on the actual car body model (see figure 5.7).

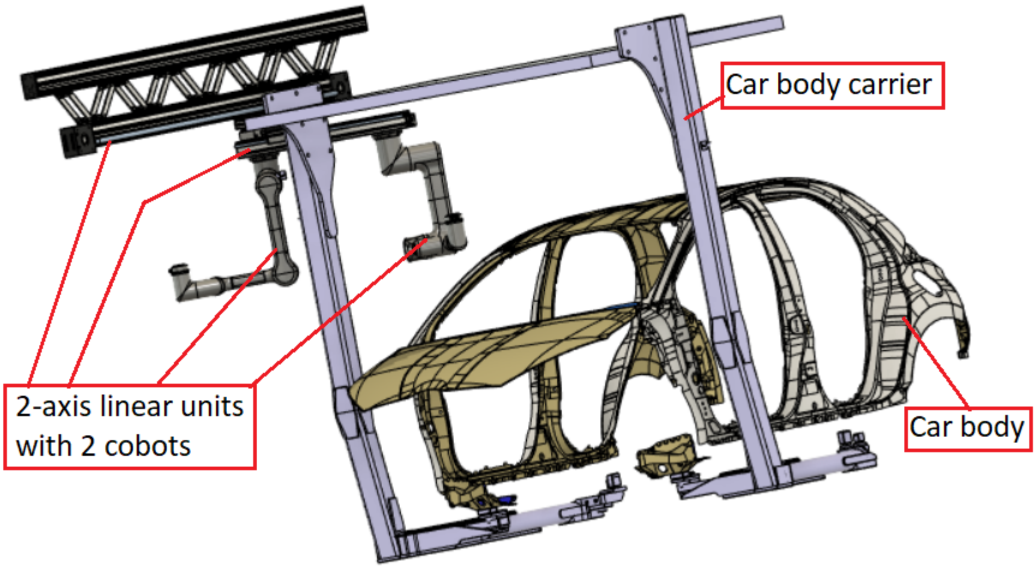


**Figure 5.7:** Final linear unit-cobot system with real car model

The chosen solution fits very well with the actual configuration of the car body with its carrier. The author will continue to add other components to this whole setup to show the overall design of the possible implementation.

## 5.4 Combining all three main components

The chosen design for modified gripper will be used to make two grippers as there are two cobots (see figure 5.8). First these grippers are connected to the linear unit-cobot system.

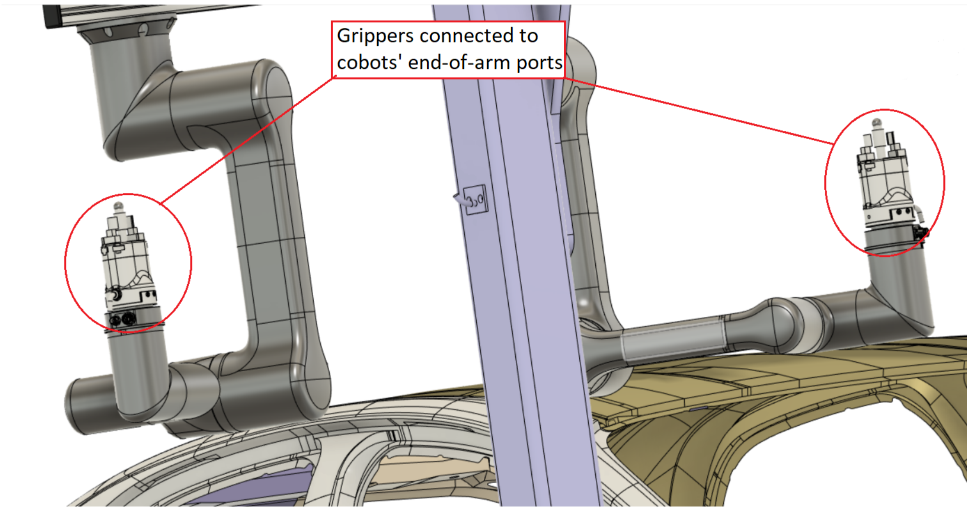


**Figure 5.8:** 2 modified grippers connected to the end-of-arm ports of 2 cobots

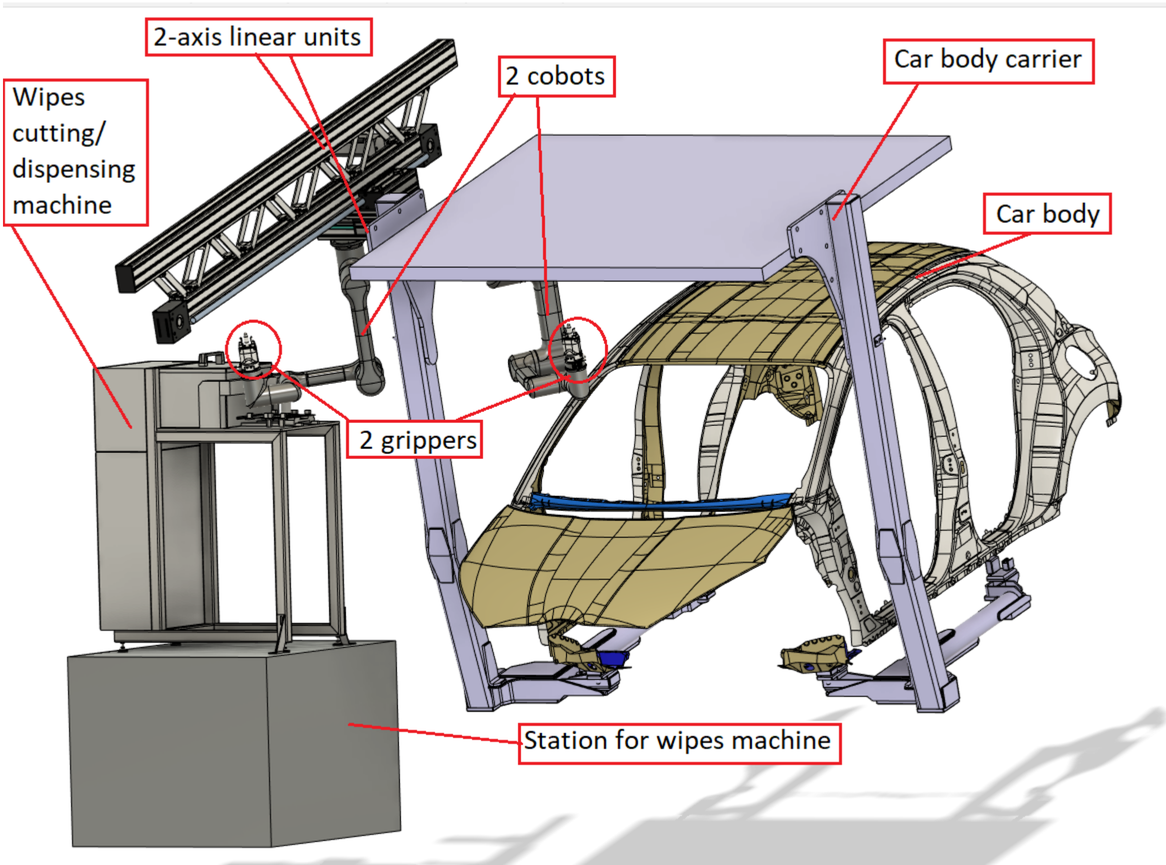


**Figure 5.9:** Full optimal solution final design

One important thing to mention here is the angle of the wipes cutting/dispensing station with the Assembly line (and linear units-cobot system) has to be 15 degrees (see figure 1.3).

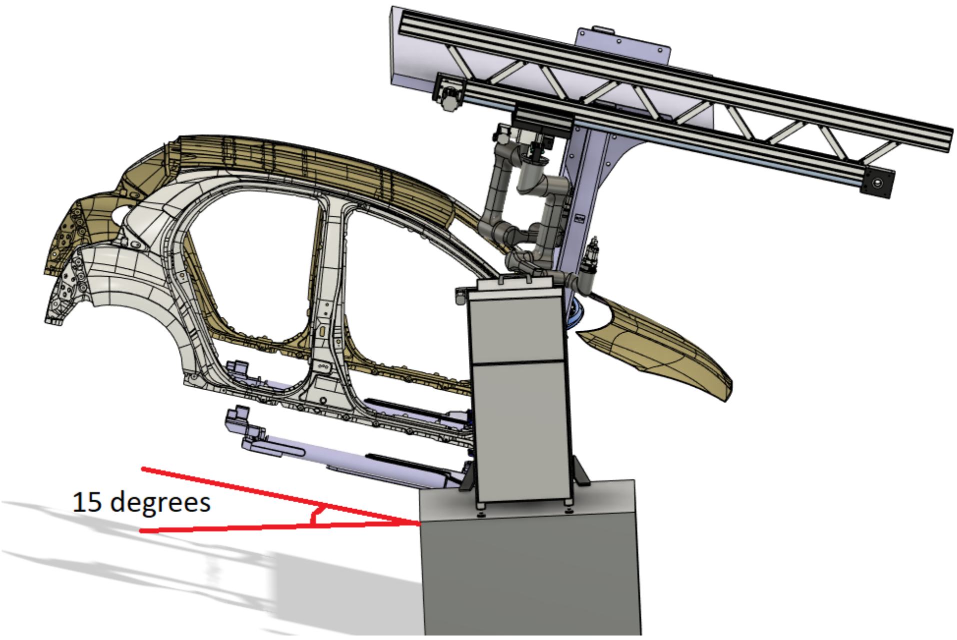


**Figure 5.10:** 15 degrees angle between wipes cutting/dispensing station and the Assembly line.

## 5.5 Description of final automated degreasing/wiping process

We can divide the complete automated process into following sub-processes:

1. **Wipes/cutting and dispensing:** the wet wipes roll is loaded into the roll compartment (see figure 5.1) with the end of the wipes fitted onto the first vacuum pad grippers. When the car body carrier crosses the reference point on the Assembly line, sending signal to the wipes machine, the first pair of vacuum pad grippers pull forward the wipe, then the linear manipulator (servo-driven, PLC controlled) moves the ultrasonic cutter in linear direction cutting the wipe with the cutting blade (see figure 4.20). The next pair of vacuum pad grippers (see figure 4.23) pull the wipe forward after cut to the platform for wipes picking by cobots.

2. **Wipes pick up by cobots:** linear units-cobot system in its home position is stationed above the wipes cutting/dispensing station. When the cut wipe comes to the pickup platform (see figure 5.3), the cobots extend the arms, then use the modified parallel grippers (see figure 5.4) to push the cut wipe into the gap of the platform while the parallel fingers grip the sides of the wet wipe, this is done by both cobots one after the other.

3. **Placing the cobots over the car body flange:** the 1st linear unit's 'slide' (see figure 3.8) follows the car body on the Assembly line in motion, at the designated point on the line, the 2nd linear unit moves towards the front of the car body carrying the 2 cobots placing them over the programmed position over the car body flange (see figure 5.7).

4. **Front windshield degreasing/wiping application:** the cobots extend their arms, the 1st cobot wipes the first half of the body flange from right side while the 2nd cobot wipes the second half of the body flange from the left side. The starting points for both cobots are in the middle, but one cobot starts from the midpoint of upper body flange while the other one starts from the midpoint of the bottom body flange, this is to avoid any collision between the grippers (see figure 6.1). So, both cobots do the wiping application either clockwise or anti-clockwise (depending on the programming) starting from the midpoints returning to the other midpoints (see figure 6.2 and 6.3).

5. **Cobots moving to back window body flange:** after the wiping application is finished on the front windshield body flange, the cobots contract their arms, then the 2nd linear unit moves away from the car body. After this, the 1st linear unit moves to the back of the car body (position determined by the length of the car body), the 2nd linear unit then moves towards the back of the car body bringing the 2 cobots over the back window body flange.

6. **Back window degreasing/wiping application:** this application is same as for the front, only difference being the size of area for degreasing/wiping. Again, the 2 cobots start from the midpoints of the upper and lower body flange and do the wiping/degreasing application either clockwise or anti-clockwise (depending on the programming), returning to the other midpoints.

7. **End of wiping/degreasing application:** After the wiping/degreasing is done for the back window, the 2nd linear unit moves away from the car body, the the cobots release the used wet wipes into another box stationed nearby on the Assembly line. Then the 1st linear unit returns to the home position, stationed over the wipes cutting/dispensing station, ready for the wiping/degreasing application of the next car.

The degreasing/wiping application will be better explained in the next chapter 6 along with a simulation done by the author (see figure 6.1).

# Chapter 6

# Simulation and programming

Having finished the final designs of the main components and the design of the automated process, the author proceeds to simulate two most important processes in this solution - the linear units-cobot system following the car body carrier and the actual wiping/degreasing application.

## 6.1 Simulation software

As discussed in the subsection of Programming in the section 3.3.2, simulation softwares are essential for the study of multiple scenarios of a robot work cell to predict the mistakes that are frequently made when designing a work cell ahead of time and for offline programming robots outside the production environment, thus, eliminating production downtime caused by shop-floor programming.[39]

The author decides to use **RoboDK**, which is one of the most commonly used robot simulation and offline programming softwares available in the industries. One of the biggest advantages of using **RoboDK** is its vast library of supported robots and cobots.

There are 5 steps to simulate and program any robot in **RoboDK**:

1. **Selection of robot:** here the user select the desired robot the vast collection robots and cobots from many manufacturers. Also, the user can add external axes such as 1, 2 or 3 axis turntables and linear rails.[39] The author chooses ***Hanwha*** manufacturer's medium-length arm cobot HCR 5A[13] and 2 external axes.

2. **Defining tool:** here the user loads a 3D model of the desired tool and convert it to a robot tool by drag and dropping it to the robot. The user also can manually enter the tool coordinates (TCP) as see in robot controller.[39]

3. **Loading 3D model of the part:** here the user loads the 3D model of

the part to be used and place it in a reference frame also known as the robot coordinate system.[39] The author uses the 3D model of the car body.

4. **Simulating toolpath:** here the user creates the paths for the robot, tool, external axes (if used) and the part to be used. The author creates the path for the ***Hanwha*** cobot, selected tool, 2 external axes and the car body 3D model.

5. **Generating programs:** here the user generates programs for the robot controller.[39] The author will later generate programs for the cobots and the 2 external axes.

## 6.2 Simulation of the degreasing/wiping application

Continuing with the final design solution (see figure 5.7), the author simulates the action of the 2 cobots hanging from the 2nd linear unit over the front body flange. (Simulation video available in the electronic attachments 8.2.)

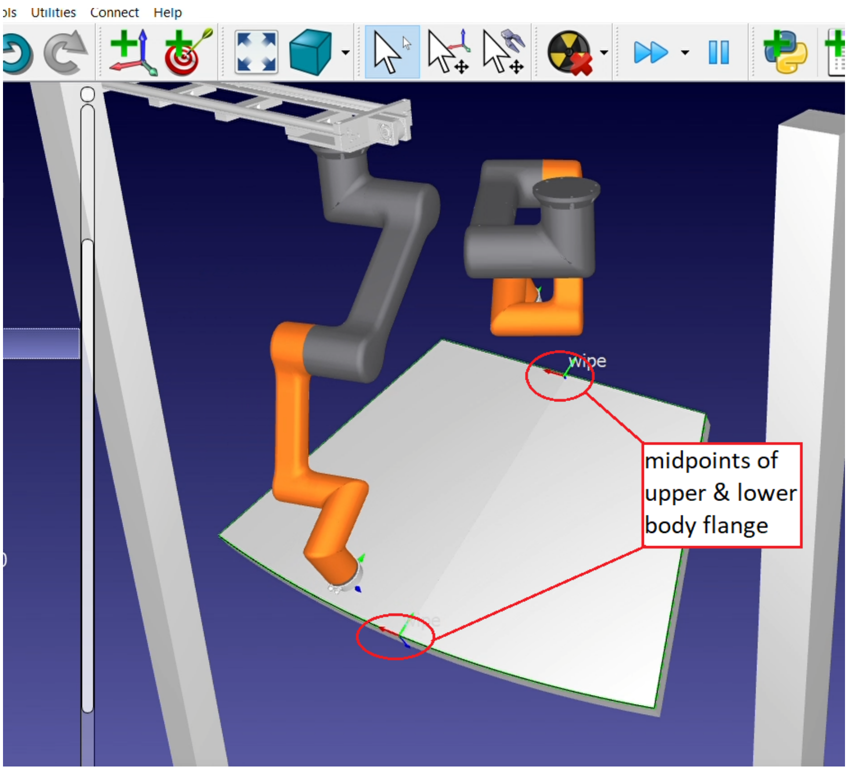


**Figure 6.1:** 2 cobots starting wiping application from the midpoints of the upper and lower body flange, simulation in RoboDK.

As seen in the simulation video (see figure 6.1), the 2 cobots start wiping/degreasing application from the midpoints of the upper and lower sections of the body flange and then going in clockwise direction around the body flange (see figures 6.2 and 6.3).

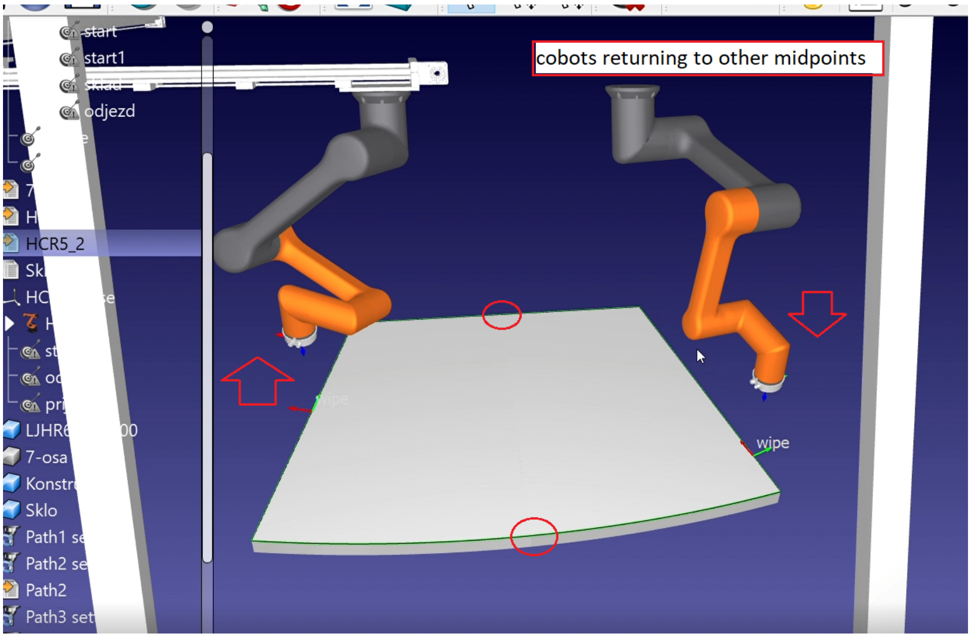


**Figure 6.2:** @ cobots wiping the body flange in clockwise direction, simulation in RoboDK.

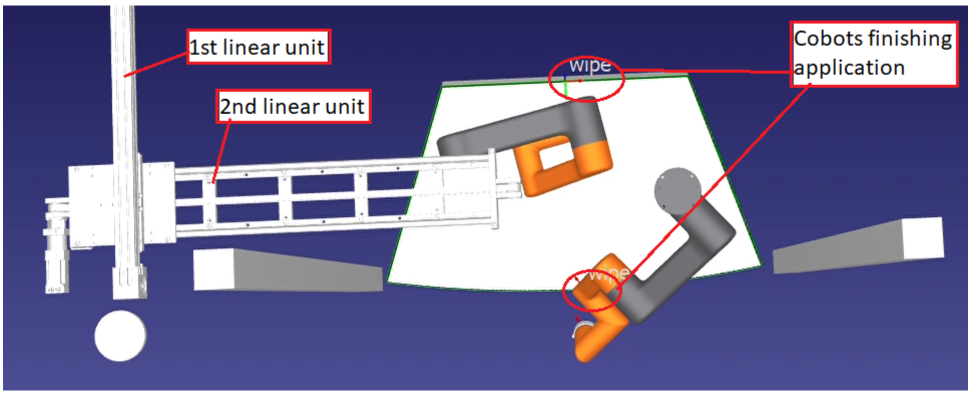


**Figure 6.3:** Top view of 2 cobots finishing wiping/degreasing application, simulation in RoboDK.

As per the simulation results, the author finalises to use 2 cobots in the same manner as displayed in the simulation, using midpoints of the upper and lower sections of the body flange as starting and ending positions which will make the overall wiping/degreasing process faster and more efficient as no single point on the body flange surface will be missed by the cobots.

## 6.3 Simulation of the linear units-cobot system

Now, the author will simulate the moving of the 2-axis linear units along with the cobots with the car body and car body carrier, all in motion as on Assembly line and also, wiping/degreasing application on the actual body flange of the 3D model of car body.(Simulation video available in the electronic attachments 8.2.)

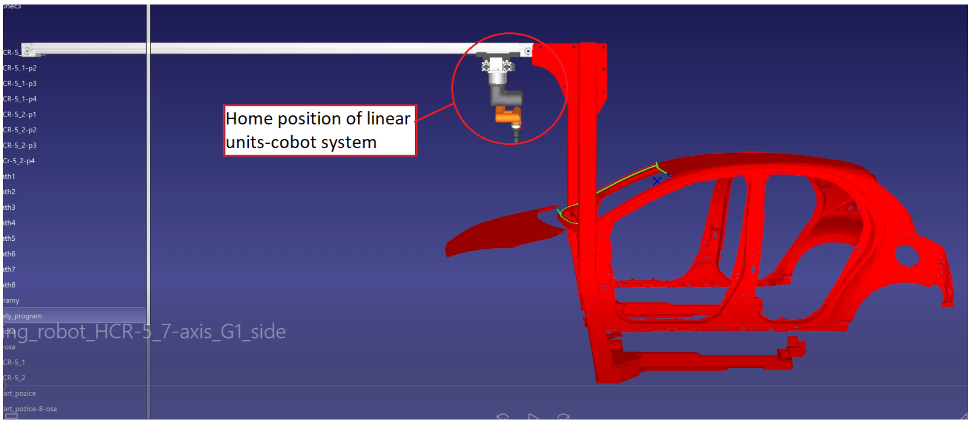


**Figure 6.4:** Side view of home position of the linear units-cobots system waiting for car body, simulation in RoboDK.

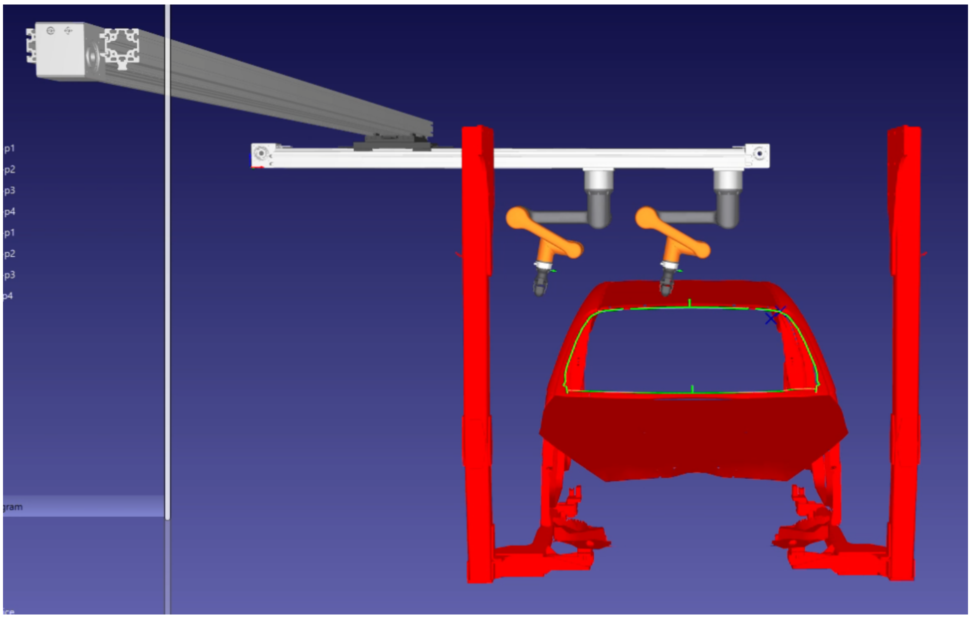

**Figure 6.5:** Front view of 2nd linear unit moving towards the car body, simulation in RoboDK.

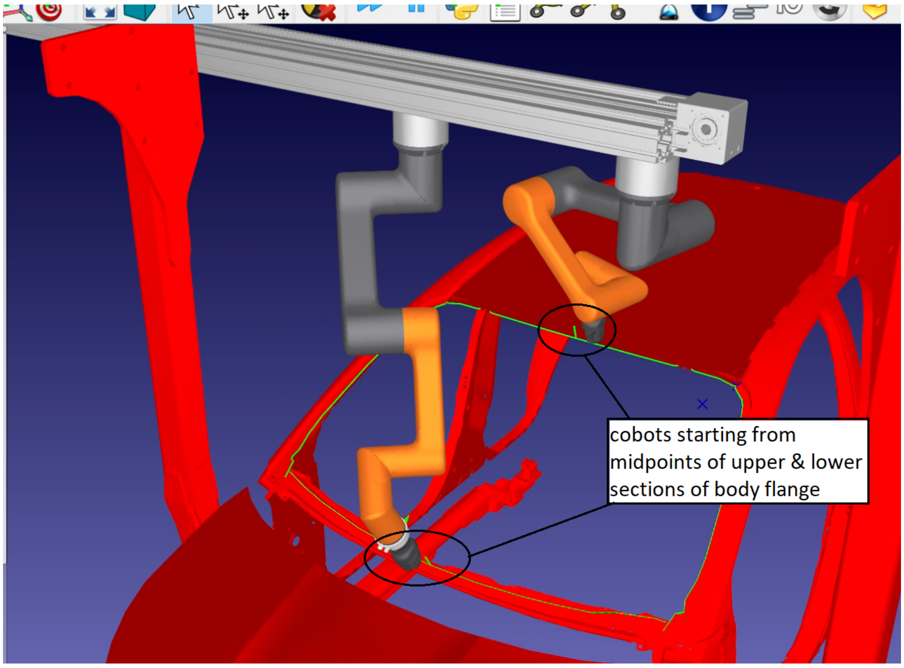


**Figure 6.6:** Cobots' wiping/degreasing application starting from the midpoints of the upper and lower sections of the body flange, simulation in RoboDK.

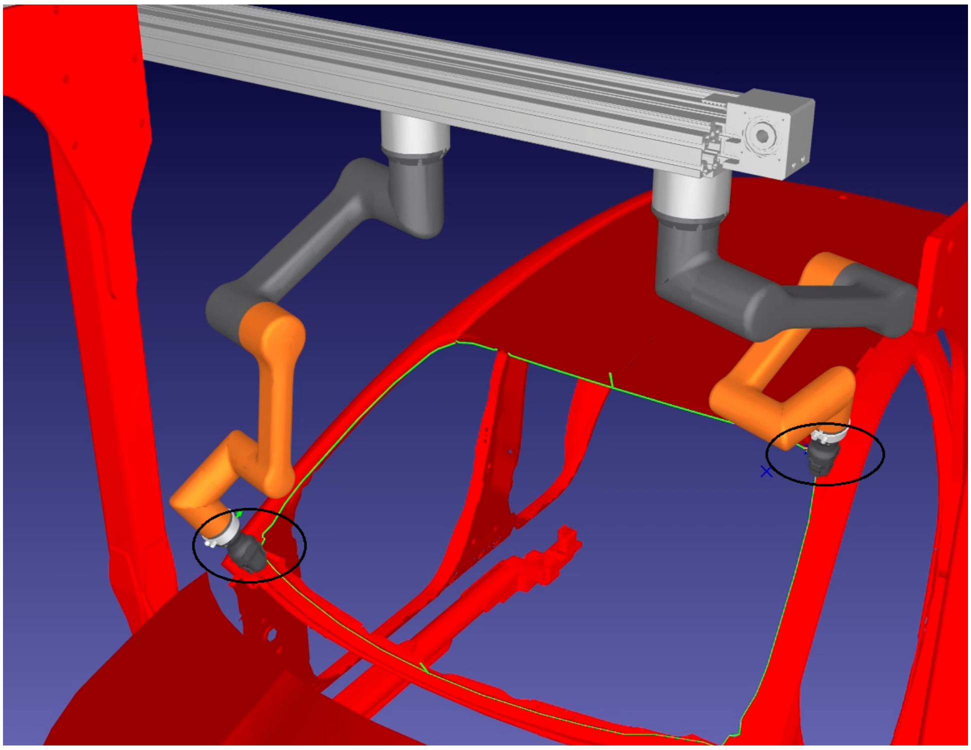

**Figure 6.7:** Cobots wiping/degreasing the opposite corners of the body flange in clockwise direction, simulation in RoboDK.

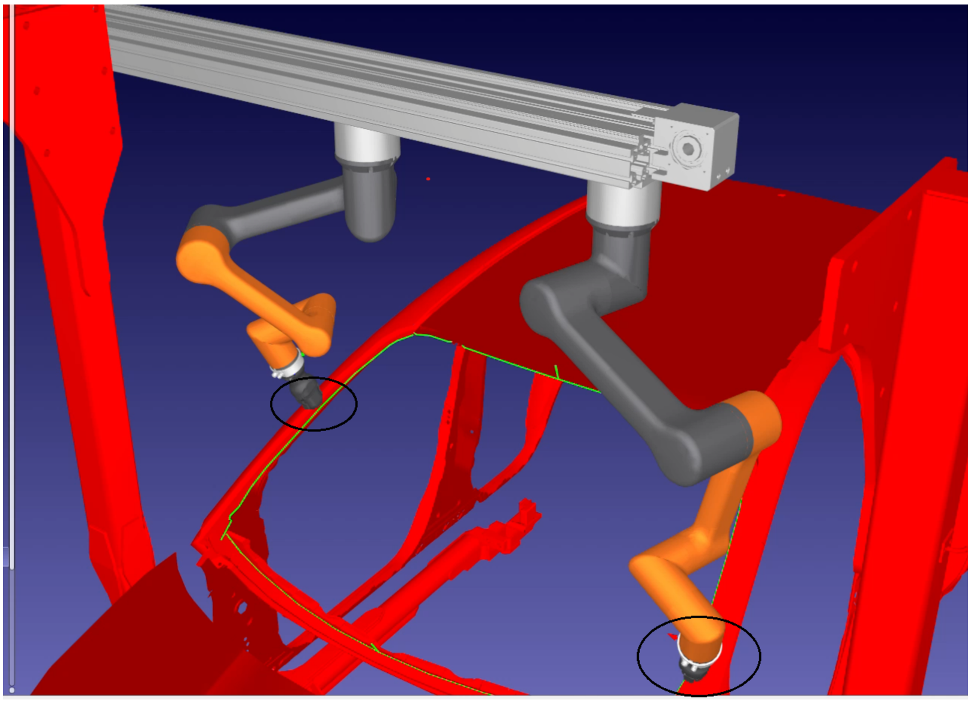

**Figure 6.8:** Half of degreasing/wiping of body flange is finished, simulation in RoboDK.

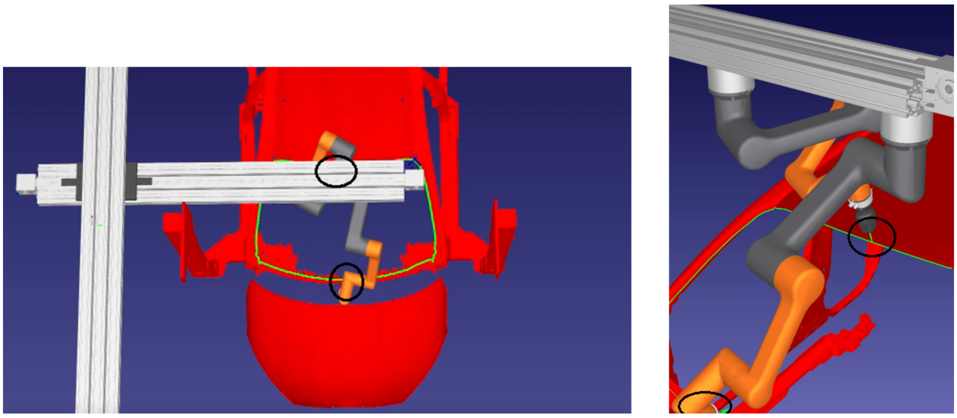

**Figure 6.9:** Degreasing/wiping of body flange by the 2 cobots is finished, simulation in RoboDK.

After the degreasing/wiping application is done for the front windshield body flange, the 2 cobots contract their arms while the 2nd linear unit moves away from the car body which then leads the 1st linear unit to go to the back of the car body where the same steps are repeated from the 2nd linear unit moving to the back of the car body leading to the 2 cobots extending their arms and doing the wiping/degreasing application on the back window body flange. After this, the whole setup returns back to the home position for the next car where the new wipes will be picked up from the wipes cutting/dispensing machine.

- Due to the wipes cutting/dispensing machine being a whole new single-purpose machine to be built in real, the simulation of cutting and dispensing wipes could not be done by RoboDK. Also, due to unavailability of the back window body flange 3D model, the author couldn't do simulation of degreasing/wiping application of the back window body flange.

## 6.4 Programming

From the simulation in RoboDK, the author was able to generate program code. (RoboDK station file (simulation file) and program codes can be found in the electronic attachments 8.)

Since the main program is divided into subprograms, the generated programs are also in multiple files. (Program code files can be found in the electronic attachments 8 or by clicking on the icon in the figure descriptions (only for electronic version).)

1. The whole program is executed by the *'Cely-program'* file (see figure ). This program starts the movement of the car body, when the movement of the car body is executed, this program creates 2 threads which execute other 2 programs - *'8-osa'* and *'7-osa'*.

```
void Cely_program(){

// *** WARNING: Invalid license or undefined post processor to generate code for Car_move***
// *** The following program will not work on the Car_move controller ***
// *** For more information:  http://www.robodk.com/help.php#PostProcessor  ***

// Program generated by RoboDK v5.3.2 for Car_move on 16/12/2021 11:53:51
// Using nominal kinematics.
DO(10) = 0;
MoveJ(Pose(-0.000000, 20.000000, -0.000000, 0.000000, 0.000000, -90.000000));
8-osa
7-osa
DI(10) = 1;
MoveJ(Pose(0.000000, -1900.000000, 200.000000, 0.000000, 0.000000, -90.000000));
MoveJ(Pose(-0.000000, 1000.000000, 200.000000, 0.000000, 0.000000, -90.000000));
return;
}
```

**Figure 6.10:** Main program named as *Cely-program* generated by RoboDK.

2. *'8-osa'* program is related to the movement of the longer linear unit denoted as 1st linear unit in the pictures before. This linear unit is responsible for following the movement of car body carrier.

```
void 8osa(){

// *** WARNING: Invalid license or undefined post processor to generate code for 8-osa***
// *** The following program will not work on the 8-osa controller ***
// *** For more information:  http://www.robodk.com/help.php#PostProcessor  ***

// Program generated by RoboDK v5.3.2 for 8-osa on 16/12/2021 12:02:23
// Using nominal kinematics.
Set_ToolFrame(Pose(350.000000, 60.000000, 130.000000, 0.000000, 0.000000, 0.000000));
MoveJ(Pose(4350.000000, 60.000000, 130.000000, 0.000000, 0.000000, -0.000000));
DO(10) = 1;
MoveJ(Joints(2950.000000));
7-od = 1;
7-od = 0;
MoveJ(Pose(4350.000000, 60.000000, 130.000000, 0.000000, 0.000000, -0.000000));
return;
}
```

**Figure 6.11:** Program named as *8-osa* for 1st linear unit's movement generated by RoboDK.

3. *'7-osa'* program is related to the movement of the shorter linear unit denoted as 2nd linear unit in the pictures before, this linear unit carries the 2 cobots. This program also creates 2 threads which execute 2 programs - *HCR5-1* and *HCR5-2*.

```
void 7osa(){

// *** WARNING: Invalid license or undefined post processor to generate code for 7-osa***
// *** The following program will not work on the 7-osa controller ***
// *** For more information:  http://www.robodk.com/help.php#PostProcessor  ***

// Program generated by RoboDK v5.3.2 for 7-osa on 16/12/2021 12:02:23
// Using nominal kinematics.
Set_ToolFrame(Pose(2450.000000, 50.000000, 10.000000, 180.000000, 0.000000, 0.000000));
MoveJ(Pose(2191.000000, 50.000000, 10.000000, 180.000000, 0.000000, 0.000000));
HCR-5_1-Done = 0;
HCR-5_2-Done = 0;
HCR-5_1
HCR-5_2
HCR-5_1-Done = 1;
HCR-5_2-Done = 1;
MoveJ(Pose(450.000000, 50.000000, 10.000000, 180.000000, 0.000000, 0.000000));
7-od = 1;
return;
}
```

**Figure 6.12:** Program named as *7-osa* for 2nd linear unit's movement generated by RoboDK.

4. HCR5-1 and HCR5-2 programs contain the actual programs related to the movements of the 2 cobots. These programs can be loaded into the robot controller and run. These programs can be modified for other car models as well. The other programs mentioned above can be substituted by the PLC controls in the real application.

```
// ----------------------
// Program HCR5_1
function HCR5_1(){
    console.log('Message: ', 'Program generated by RoboDK v5.3.2 for HCR-5_1 on 16/12/2021 12:02:23');
    console.log('Message: ', 'Using nominal kinematics.');
    setToolCenterPoint({x: 0.000, y: 0.000, z: 153.000, rx: -0.000, ry: 0.000, rz: -0.000});
    // setPayload(0.5, {x:0, y:0, z:150}); // sample 0.5kg, CoG is (0,0,150)
    setGeneralDigitalOutput(HCR-5_1-Foldlm1, 0);
    Path1()
    Path2()
    Path3()
    Path4()
    moveJoint([118.479000, 9.050780, -133.430000, 51.824700, 97.819400, -207.066000], 500.0, 800.0);
    setGeneralDigitalOutput(HCR-5_1-Foldlm1, 1);
    moveJoint([0.000000, 0.000000, -150.000000, 70.000000, 90.000000, -0.000000], 500.0, 800.0);
    setGeneralDigitalOutput(HCR-5_1-Done, 1);
}
// End of program HCR5_1


// ----------------------
// Program Path1
function Path1(){
    console.log('Message: ', 'Show RE_wipe1');
    moveJoint([8.421960, 11.536400, -139.889000, 42.503000, 71.674000, -99.298700], 500.0, 800.0);
    moveLinear('tcp', {x: -336.000, y: -328.750, z: 636.140, rx: 18.734, ry: -1.255, rz: -180.000}, 1000.0, 1000.0);
    moveLinear('tcp', {x: -180.335, y: -333.530, z: 638.120, rx: 18.752, ry: -1.256, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -25.403, y: -348.280, z: 644.220, rx: 18.668, ry: -3.871, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -20.629, y: -350.580, z: 643.930, rx: 18.544, ry: -5.265, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -15.883, y: -352.870, z: 643.640, rx: 18.488, ry: -5.346, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -13.904, y: -353.140, z: 643.760, rx: 18.466, ry: -5.405, rz: 180.000}, 300.0, 1000.0);
}
// End of program Path1


// ----------------------
// Program Path2
function Path2(){
    console.log('Message: ', 'Show RE_wipe1');
    moveJoint([63.061000, -21.866300, -137.839000, 88.887500, 87.289900, -161.374000], 500.0, 800.0);
    moveLinear('tcp', {x: 29.650, y: -363.120, z: 645.120, rx: 18.411, ry: -6.116, rz: -164.475}, 300.0, 1000.0);
    moveLinear('tcp', {x: 32.556, y: -362.450, z: 645.670, rx: 18.425, ry: -6.168, rz: -164.535}, 300.0, 1000.0);
    moveLinear('tcp', {x: 32.896, y: -361.350, z: 646.070, rx: 18.468, ry: -6.248, rz: 174.053}, 300.0, 1000.0);
    moveLinear('tcp', {x: 97.349, y: -370.200, z: 651.300, rx: 18.495, ry: -6.893, rz: 177.397}, 300.0, 1000.0);
    moveLinear('tcp', {x: 111.941, y: -371.200, z: 653.010, rx: 18.473, ry: -7.591, rz: -171.868}, 300.0, 1000.0);
    moveLinear('tcp', {x: 144.859, y: -366.470, z: 659.510, rx: 18.683, ry: -8.084, rz: 160.059}, 300.0, 1000.0);
    moveLinear('tcp', {x: 156.977, y: -371.260, z: 659.770, rx: 18.692, ry: -8.599, rz: 160.059}, 300.0, 1000.0);
}
// End of program Path2


// ----------------------
// Program Path3
function Path3(){
    console.log('Message: ', 'Show RE_wipe1');
    moveJoint([98.847100, -13.073800, -130.023000, 70.812100, 101.914000, -188.533000], 500.0, 800.0);
    moveLinear('tcp', {x: 203.018, y: -322.080, z: 684.520, rx: 19.398, ry: -8.992, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 214.432, y: -302.760, z: 693.750, rx: 20.077, ry: -9.581, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 214.416, y: -302.840, z: 693.820, rx: 21.728, ry: -10.140, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 250.879, y: -192.050, z: 745.050, rx: 21.639, ry: -10.429, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 276.815, y: -74.970, z: 803.490, rx: 24.105, ry: -11.588, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 311.590, y: 116.520, z: 909.900, rx: 26.839, ry: -13.260, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 316.175, y: 173.450, z: 942.710, rx: 28.863, ry: -13.903, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 311.219, y: 225.890, z: 971.120, rx: 29.608, ry: -14.120, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 311.492, y: 229.060, z: 973.030, rx: 29.936, ry: -14.118, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 320.632, y: 253.380, z: 989.910, rx: 30.099, ry: -14.435, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 304.281, y: 265.350, z: 991.980, rx: 30.415, ry: -13.939, rz: -180.000}, 1000.0, 1000.0);
    moveLinear('tcp', {x: 295.258, y: 316.410, z: 1019.560, rx: 30.415, ry: -13.939, rz: -180.000}, 300.0, 1000.0);
}
// End of program Path3


// ----------------------
// Program Path4
function Path4(){
    console.log('Message: ', 'Show RE_wipe1');
    moveJoint([214.915000, -30.315700, -52.045700, -37.062700, 106.066000, -342.804000], 500.0, 800.0);
    moveLinear('tcp', {x: 256.946, y: 315.550, z: 1008.610, rx: 30.032, ry: -12.967, rz: 150.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 251.059, y: 317.560, z: 1008.180, rx: 30.032, ry: -12.967, rz: 150.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 250.562, y: 317.770, z: 1008.170, rx: 30.032, ry: -12.967, rz: 150.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 250.553, y: 317.780, z: 1008.170, rx: 30.032, ry: -12.967, rz: 150.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 163.596, y: 352.610, z: 1006.990, rx: 30.096, ry: -11.199, rz: 150.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 110.263, y: 381.250, z: 1012.510, rx: 30.542, ry: -9.183, rz: 170.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 22.192, y: 406.880, z: 1011.610, rx: 30.208, ry: -7.585, rz: 170.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -66.815, y: 427.170, z: 1011.180, rx: 29.925, ry: -5.328, rz: 170.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -200.264, y: 446.540, z: 1011.080, rx: 30.260, ry: -2.495, rz: -170.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -336.000, y: 452.890, z: 1010.790, rx: 29.645, ry: 0.349, rz: -150.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -336.305, y: 477.620, z: 967.330, rx: 29.645, ry: 0.349, rz: -150.000}, 1000.0, 1000.0);
}
// End of program Path4
```

**Figure 6.13:** Program named as *HCR5-1* for 1st cobot's movement generated by RoboDK.

```
// -----------------------
// Program HCR5_2
function HCR5_2(){
    console.log('Message: ', 'Program generated by RoboDK v5.3.2 for HCR-5_2 on 16/12/2021 12:02:25');
    console.log('Message: ', 'Using nominal kinematics.');
    setToolCenterPoint({x: 0.000, y: 0.000, z: 153.000, rx: -0.000, ry: 0.000, rz: -0.000});
    // setPayload(0.5, {x:0, y:0, z:150}); // sample 0.5kg, CoG is (0,0,150)
    Path5()
    Path6()
    Path7()
    Path8()
    // wait(getGeneralDigitalInput(0) == 1);
    moveJoint([0.000000, 0.000000, -150.000000, 70.000000, 90.000000, 0.000000], 500.0, 800.0);
    setGeneralDigitalOutput(HCR-5_2-Done, 1);
}
// End of program HCR5_2


// -----------------------
// Program Path5
function Path5(){
    console.log('Message: ', 'Show RE_wipe2');
    moveJoint([-131.665000, -32.211200, -25.978200, -53.915200, 110.770000, 38.039500], 500.0, 800.0);
    moveLinear('tcp', {x: 334.000, y: 452.850, z: 1010.850, rx: 29.922, ry: 1.777, rz: -180.000}, 1000.0, 1000.0);
    moveLinear('tcp', {x: 198.602, y: 446.610, z: 1011.020, rx: 30.002, ry: 4.618, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 65.146, y: 427.210, z: 1011.210, rx: 30.070, ry: 7.965, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -99.788, y: 385.250, z: 1012.320, rx: 30.321, ry: 11.078, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -192.700, y: 352.720, z: 1013.480, rx: 30.305, ry: 12.257, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -284.367, y: 313.150, z: 1014.490, rx: 30.415, ry: 13.939, rz: 180.000}, 300.0, 1000.0);
}
// End of program Path5


// -----------------------
// Program Path6
function Path6(){
    console.log('Message: ', 'Show RE_wipe2');
    moveJoint([-67.833500, -30.794900, -47.509500, -44.757100, 92.931700, -19.076900], 500.0, 800.0);
    moveLinear('tcp', {x: -306.250, y: 265.300, z: 992.090, rx: 30.104, ry: 14.651, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -316.004, y: 263.190, z: 993.720, rx: 30.104, ry: 14.651, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -322.622, y: 253.360, z: 989.940, rx: 30.104, ry: 14.651, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -322.628, y: 253.390, z: 989.930, rx: 29.770, ry: 14.509, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -313.481, y: 229.060, z: 973.080, rx: 29.435, ry: 14.368, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -313.215, y: 225.900, z: 971.150, rx: 29.103, ry: 14.225, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -318.174, y: 173.460, z: 942.720, rx: 28.437, ry: 13.939, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -313.609, y: 116.570, z: 909.830, rx: 26.586, ry: 12.772, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -278.830, y: -74.930, z: 803.430, rx: 23.760, ry: 11.174, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -252.873, y: -192.080, z: 745.070, rx: 22.446, ry: 10.602, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -216.422, y: -302.760, z: 693.810, rx: 19.604, ry: 9.920, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -216.427, y: -302.740, z: 693.790, rx: 19.282, ry: 9.766, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -205.001, y: -322.100, z: 684.630, rx: 18.960, ry: 9.612, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -169.547, y: -351.000, z: 668.360, rx: 18.530, ry: 8.856, rz: -180.000}, 300.0, 1000.0);
}
// End of program Path6


// -----------------------
// Program Path7
function Path7(){
    console.log('Message: ', 'Show RE_wipe2');
    moveJoint([23.178600, -6.331630, -137.968000, 53.540600, 68.885100, -111.538000], 500.0, 800.0);
    moveLinear('tcp', {x: -151.924, y: -362.800, z: 661.440, rx: 19.069, ry: 8.029, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -111.838, y: -366.480, z: 654.210, rx: 18.939, ry: 7.612, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -100.025, y: -365.470, z: 652.900, rx: 18.872, ry: 6.931, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -36.441, y: -356.730, z: 647.770, rx: 18.586, ry: 6.216, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -30.580, y: -358.070, z: 646.650, rx: 18.562, ry: 6.130, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -27.658, y: -358.740, z: 646.060, rx: 18.733, ry: 6.154, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: -25.684, y: -358.450, z: 645.940, rx: 18.733, ry: 6.154, rz: -180.000}, 300.0, 1000.0);
}
// End of program Path7


// -----------------------
// Program Path8
function Path8(){
    console.log('Message: ', 'Show RE_wipe2');
    moveJoint([45.872500, -26.320000, -142.808000, 89.144000, 72.910500, -136.467000], 500.0, 800.0);
    moveLinear('tcp', {x: 13.883, y: -352.840, z: 643.560, rx: 19.001, ry: 5.589, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 18.629, y: -350.550, z: 643.850, rx: 19.043, ry: 5.500, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 23.403, y: -348.260, z: 644.170, rx: 18.941, ry: 3.892, rz: -180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 178.335, y: -333.530, z: 638.090, rx: 18.903, ry: 1.260, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 334.000, y: -328.730, z: 636.090, rx: 19.028, ry: 1.263, rz: 180.000}, 300.0, 1000.0);
    moveLinear('tcp', {x: 332.898, y: -312.430, z: 588.840, rx: 19.028, ry: 1.263, rz: 180.000}, 1000.0, 1000.0);
}
// End of program Path8
```

**Figure 6.14:** Program named as *HCR5-2* for 2nd cobot's movement generated by RoboDK.

As the programming for the linear units and cobots' movements is done, the remaining programming is left for the gripper, wipes cutting/dispensing machine and the connection with the main control communications system used at **TMMCZ**.

1. Regarding gripper, programming depends on the manufacturer of the gripper (such as Robotiq, OnRobot, DH Robotics) and manufacturer of the cobot (such as Hanwha, Kuka, Doosan, FANUC). As the author used *Hanwha* cobots, the gripper can be programmed by the software provided by *Hanwha* called ***RODi***.[12] The gripper file (in the format .asar) is added for the chosen gripper manufacturer and can then be configured and programmed along with the cobot.

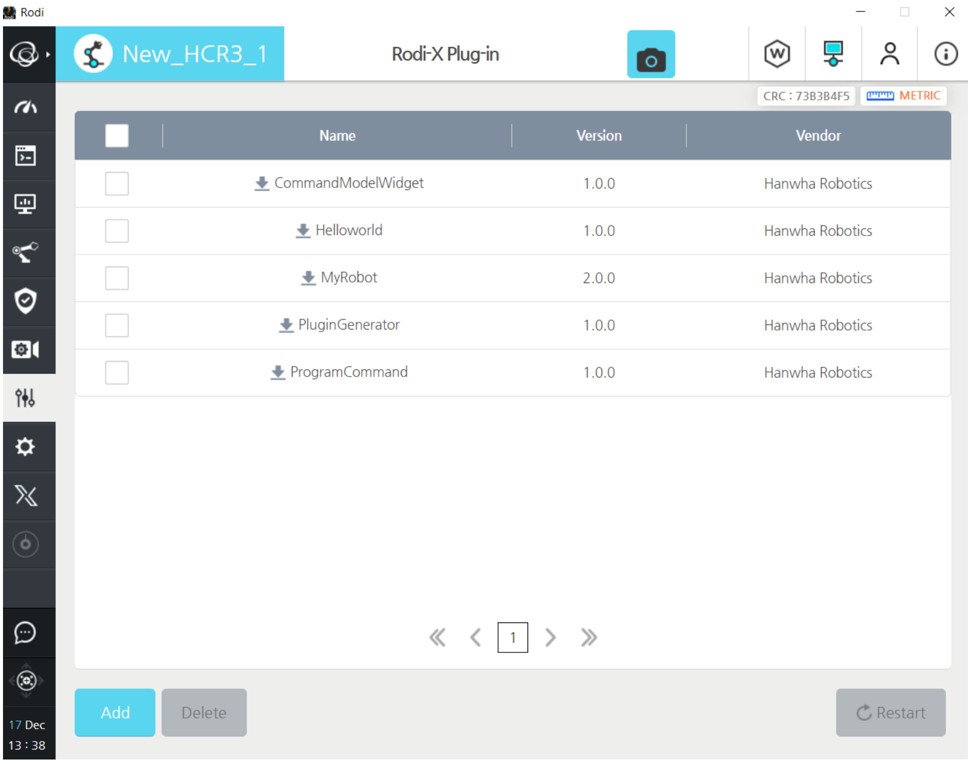


**Figure 6.15:** Menu for adding gripper configuration file in RODi software.[12]

2. The programming and control of the wipes cutting/dispensing machine is proposed by the author to be done using PLC (Programmable Logic Controller) controls and HMI (Human Machine Interface) to connect with the 1st linear unit and the cobots so cut wipes only when required by the cobots.

**Figure 6.16:** PLC and HMI from Omron manufacturer.[27]

3. The central control communications system used at **TMMCZ** is ***TOYOPUC PC10 series***[60]. So, the 1st linear unit, cobots and the wipes cutting/dispensing machine should be connected to this central system to avoid any collision or body damage in case of sudden line stop or human interference. Also, this central system connection can be used to identify car model when degreasing/wiping different car models. The 2nd linear unit doesn't need to be connected to this central system as it relies on 1st linear unit's movement to move itself.

**Figure 6.17:** TOYOPUC PC10 series programmable controller.[45]

## Chapter 7

# Conclusion

In the course of this work, the author was able to design the main equipment as well as a complete process for the automation of the given process of manual degreasing/wiping of the car body flange. The author was also able to simulate the important automated processes to verify the feasibility of the whole solution proposed. So, the objectives of this thesis were achieved by the author.

The author was also able to study and understand the concepts of industrial automation technologies and their benefits which will prove very useful to the author's current job as Automation engineer at TMMCZ Kolin for future projects.

This thesis can serve as the base study to be given to TMMCZ's automation technology partners to be further expanded to implement this solution in real-life application which will reduce the potential health and safety risks the operators are currently facing in the degreasing process while also improving the overall quality of the process as all car bodies will be consistently degreased without any factor of potential human error. The designs proposed by this thesis can serve as the base designs for real equipment to be made by automation development companies. Furthermore, the study can be expanded for different car models by adding more programs for the cobots for each car model's body flange.

This project can also be a substantial inspiration in bringing Industry 4.0 to TMMCZ Kolin meaning opening doors for possibilities of implementing such automation technologies in more processes to increase efficiency and quality of the products in the end.

# Chapter 8

# List of Electronic Attachments

## 8.1 Design files

The following .step files can be opened by any 3D viewer or editor applications.

1. 1st idea.step
2. 2nd idea.step
3. Single-axis linear unit with cobot hanging down.step
4. Single-axis linear unit with cobot standing above.step
5. 2-sided single-axis linear units with cobots.step
6. 2-axis linear units and 2 cobots with real car body carrier.step
7. Cobot gripper with nozzles for liquid alcohol and comp air.step
8. 2-finger gripper with nozzle for liq alcohol.step
9. DH Robotics parallel gripper 3D model with foam middle piece.step
10. DH Robotics parallel gripper with foam blocks on fingers.step
11. First design idea for wipes cutting/dispensing machine.step
12. Final design for wipes cutting/dispensing machine.step
13. Final design for modified cobot gripper.step
14. Final linear unit-cobot system with real car model.step
15. Full optimal solution final design.step

## 8.2 Video files

1. 1st-Test-HennLich.mp4
2. 2nd-Test-HennLich.mp4
3. Ultrasonic-cutting-test-DJK.mp4
4. Wiping-application-cobots.mp4
5. Front-view-of-full-system-simulation.mp4
6. Normal-view-of-full-system-simulation.mp4
7. Side-view-of-full-system-simulation.mp4
8. Top-view-of-full-system-simulation.mp4

## 8.3 Simulation files

1. Station-cobot-linear-unit-car-body.rdk (to be opened in RoboDK software only)

## 8.4 Program files

1. Cely-program.txt
2. 8-osa.txt
3. 7-osa.txt
4. HCR5-1.js
5. HCR5-2.js

# Bibliography


[1] 3-finger adaptive robot gripper - robotiq. https://robotiq.com/products/3-finger-adaptive-robot-gripper. (Accessed on 01/05/2022).

[2] Amazon web services - wikipedia. https://en.wikipedia.org/wiki/Amazon_Web_Services. (Accessed on 12/29/2021).

[3] Belt drive or lead screw? the answer is in the application. https://www.pbclinear.com/Blog/2020/February/Lead-Screw-or-Belt-Drives. (Accessed on 21/12/2021).

[4] Cobot basics: 5 differences between collaborative robots and traditional robots | blog | industrial equipment supplier - elixir philippines. https://www.elixirphil.com/cobot-basics-differences-collaborative-robots-traditional-robots/. (Accessed on 01/01/2022).

[5] Cobots vs. industrial robots: what are the differences? https://www.cobottrends.com/cobots-vs-industrial-robots-what-are-differences/. (Accessed on 01/02/2022).

[6] The dangers of isopropyl alcohol (ipa) - flammability, exposure, & safety. https://blog.gotopac.com/2016/01/06/the-dangers-of-isopropyl-alcohol/#:~:text=Isopropyl%20alcohol%20is%20readily%20absorbed,skin%20contact%20may%20cause%20corrosion. (Accessed on 01/11/2021).

[7] Dh-robotics | pgc. https://en.dh-robotics.com/pgc/. (Accessed on 01/05/2022).

[8] Dual quick changer - improves cycle time and productivity | onrobot. https://onrobot.com/en/products/dual-quick-changer?_gl=1*faok6d*_ga*MTAwNTYwMDI3OS4xNjQxMzM0MjEw*_up*MQ... (Accessed on 01/04/2022).

[9] Foam suction cup. http://www.acy.com.tw/Foam-Suction-Cup. (Accessed on 01/09/2022).

[10] Fusion 360 | 3d cad, cam, cae & pcb cloud-based software | autodesk. https://www.autodesk.com/products/fusion-360/overview. (Accessed on 01/03/2022).

[11] Hanwha corporation/machinery. https://www.hanwharobotics.com/En/Product/HCR12. (Accessed on 01/03/2022).

[12] Hanwha corporation/machinery. https://www.hanwharobotics.com/En/Product/Product. (Accessed on 01/09/2022).

[13] Hanwha corporation/machinery. https://www.hanwharobotics.com/En/Product/HCR5. (Accessed on 01/09/2022).

[14] Hex 6 axis force torque sensor | onrobot. https://onrobot.com/en/products/hex-6-axis-force-torque-sensor. (Accessed on 01/05/2022).

[15] Industrial robot - wikipedia. https://en.wikipedia.org/wiki/Industrial_robot. (Accessed on 28/12/2021).

[16] Industrial robots. https://www.mhi.org/fundamentals/robots. (Accessed on 28/12/2021).

[17] Industry 4.0 and the fourth industrial revolution explained. https://www.i-scoop.eu/industry-4-0/. (Accessed on 12/27/2021).

[18] The item automation system: an end-to-end automation solution - item. https://www.item24.de/en/productworld/automation/. (Accessed on 24/12/2021).

[19] Jumbo roll cross lapping/parallel spunlace nonwoven fabric for wet wipes - buy cleaning wipes nonwoven fabric,nonwoven fabric for wet wipes,printed nonwoven fabric product on alibaba.com. https://www.alibaba.com/product-detail/Jumbo-roll-Cross-Lapping-Parallel-spunlace_60411079480.html. (Accessed on 01/07/2022).

[20] Karakuri/lca - start. https://karakuri.item24.de/. (Accessed on 11/12/2021).

[21] Karakuri/lca for transporting materials in small car production - item - blog. https://blog.item24.com/en/lean-production/karakuri-lca-for-transporting-materials-in-small-car-production/. (Accessed on 01/02/2022).

[22] Karakuri/lca in a whole new dimension! - youtube. https://www.youtube.com/watch?v=fF-Gc5_8Ytc&ab_channel=item24com. (Accessed on 01/02/2022).

[23] Linear hct7c 7ft chain drive t-rail for ldo33, ldo50, lso50, lco75 operators | ebay. https://www.ebay.com/itm/323624888252. (Accessed on 22/12/2021).

[24] Linear recirculating ball bearing units | schaeffler turkey. https://www.schaeffler.com/content.schaeffler.tr/en/products-and-solutions/industrial/product-portfolio/linear_guidance_systems/linear_recirculating_ball_bearing_units/index.jsp. (Accessed on 21/12/2021).

[25] Linear technology – the simple solution for automation - item - blog. https://blog.item24.com/en/automation/linear-technology-the-simple-solution-for-automation/. (Accessed on 01/02/2022).

[26] The modern industrial workhorse: Pid controllers - tech briefs. https://www.techbriefs.com/component/content/article/tb/pub/features/articles/20013. (Accessed on 11/22/2021).

[27] Omron cp1e digital hmi plc trainer – plc-trainer. http://plc-trainer.com/p/omron-cp1e-digital-hmi-plc-trainer/. (Accessed on 01/09/2022).

[28] Onrobot quick changer | onrobot. https://onrobot.com/cs/node/1637. (Accessed on 01/05/2022).

[29] An overview on industrial automation - structures & technologies. https://www.elprocus.com/an-overview-on-industrial-automation/. (Accessed on 11/22/2021).

[30] Parallel gripper for robot arms - eoat | onrobot. https://onrobot.com/en/products/2fg7. (Accessed on 01/05/2022).

[31] Productdetails | item industrietechnik gmbh. https://product.item24.de/en/products/product-catalogue/productdetails/products/linear-units-with-a-chain-drive-1001374071/linear-unit-lre-8-d14-80x40-ku-80-66454/. (Accessed on 22/12/2021).

[32] Rack and pinion drive - lifgo® linear - leantechnik ag. https://www.directindustry.com/prod/leantechnik-ag/product-56962-1554705.html. (Accessed on 22/12/2021).

[33] Rack and pinion drive system: What is it? - motion control tips. https://www.motioncontroltips.com/what-is-a-rack-and-pinion/. (Accessed on 22/12/2021).

[34] Rg2 gripper - flexible 2 finger robot gripper | onrobot. https://onrobot.com/en/products/rg2-gripper. (Accessed on 01/05/2022).

[35] Robot software - wikipedia. https://en.wikipedia.org/wiki/Robot_software#Examples_of_programming_languages_for_Industrial_Robots. (Accessed on 30/12/2021).

[36] Robotika: špičková kvalita hennlich. https://www.hennlich.cz/produkty/robotika-13509.html. (Accessed on 01/05/2022).

[37] Roller rail guides: Three reasons why you should use them. https://www.linearmotiontips.com/three-reasons-use-roller-rail-guides/. (Accessed on 21/12/2021).

[38] Seventh axis for cobots: top quality hennlich. https://www.hennlich.cz/produkty/robotika-vse-pro-koboty-14085/sedma-osa-pro-koboty.html. (Accessed on 01/03/2022).

[39] Simulator for industrial robots and offline programming - robodk. https://robodk.com/. (Accessed on 01/09/2022).

[40] Sliding linear guides | bearings | oiles. https://www.oiles.co.jp/en/bearing/slide_shifter/sliding-linear-guide/. (Accessed on 12/12/2021).

[41] Soft gripper - flexible, food-grade robot gripper | onrobot. https://onrobot.com/en/products/soft-gripper. (Accessed on 01/05/2022).

[42] Square balancer|endo. http://www.endo-kogyo.co.jp/english/product/karakuri/square/index.html. (Accessed on 11/12/2021).

[43] T-slot slider 8 80x40 | alváris profile systems. https://www.alvaris.eu/en/building-kit-systems-aluminumprofiles-fasteners-accessories/accessories-range-8/sliding-guides/t-slot-slider-8-80x40. (Accessed on 21/12/2021).

[44] Top 12 facts about ce marking of electrical and electronic equipment - eurofins e&e north america. https://metlabs.com/regulation/top-12-facts-about-ce-marking-of-electrical-and-electronic-equipment/. (Accessed on 01/05/2022).

[45] Toyopuc pc10 seriesjtekt corporation. https://toyoda.jtekt.co.jp/e/products/products-list00_02.html. (Accessed on 01/09/2022).

[46] Toyota motor manufacturing czech republic s.r.o. https://www.toyotacz.com/en/. (Accessed on 12/30/2021).

[47] Ultrasonic cutter / djk europe. https://djkeurope.com/en/business-fields/automotive/ultrasonic-cutting/information-about-ultrasonic-cutting-technology/. (Accessed on 01/09/2022).

[48] Ultrasonic cutter qualityultrasonic cutter and ultrasonic polisher columnssonotec. https://www.sonotec.com/en/column/cutting_quality.html. (Accessed on 01/09/2022).

[49] Vacuum grippers - robotiq. https://robotiq.com/products/vacuum-grippers. (Accessed on 01/05/2022).

[50] Vg10 - electrical vacuum gripper with suction cups | onrobot. https://onrobot.com/en/products/vg10-electric-vacuum-gripper. (Accessed on 01/05/2022).

[51] The vgc10 electric vacuum gripper - small, but powerful | onrobot. https://onrobot.com/en/products/vgc10. (Accessed on 01/05/2022).

[52] What is automation? - definition from techopedia. https://www.techopedia.com/definition/32099/automation. (Accessed on 10/17/2021).

[53] What is automation? - isa. https://www.isa.org/about-isa/what-is-automation. (Accessed on 10/17/2021).

[54] Who needs the internet of things? - linux.com. https://www.linux.com/news/who-needs-internet-things/. (Accessed on 12/29/2021).

[55] Yescom 28" vinyl cutter machine 720mm sign cutting plotter with signmaster software 3 blades adjustable force speed - walmart.com. https://www.walmart.com/ip/Yescom-28-Vinyl-Cutter-Machine-720mm-Sign-Cutting-Plotter-with-Signmaster-Software-998525507. (Accessed on 01/07/2022).

[56] Česko / djk europe. https://djkeurope.com/en/cesko/. (Accessed on 01/09/2022).

[57] AssemblyErgo. Final 1 degreasing ergo, 2021.

[58] David Balderas, Alexandro Ortiz, Efraín Méndez, Pedro Ponce, and Arturo Molina. Empowering digital twin for industry 4.0 using metaheuristic optimization algorithms: case study pcb drilling optimization. *The International Journal of Advanced Manufacturing Technology*, 113, 03 2021.

[59] Armando W. Colombo and Robert Harrison. Modular and collaborative automation: achieving manufacturing flexibility and reconfigurability. *International Journal of Manufacturing Technology and Management*, 14(3-4):249–265, 2008.

[60] Process Engineering. Communications systems. *TMMCZ Process Engineering*, 2008.

[61] Process Engineering. Worker ergonomics. *TMMCZ Process Design*, 1 2021.

[62] Production Engineering. Final 1 layout. *TMMCZ Assembly Layouts*, 2021.

[63] Madhab C. Jena, Sarat K. Mishra, and Himanshu S. Moharana. Application of industry 4.0 to enhance sustainable manufacturing. *Environmental Progress & Sustainable Energy*, 39(1):13360, 2020.

[64] Henning Kagermann, W Lukas, and Wolfgang Wahlster. Abschotten ist keine alternative. *VDI nachrichten*, 16, 2015.

[65] Suyash Kashyap. Cost reduction proposal for degreasing process. *TMMCZ CRA*, 2, 2021.

[66] John E Kelly III. Computing, cognition and the future of knowing: How humans and machines are forging a new age of understanding. *IBM Research and Solutions Portfolio*, pages 2–11, 2015.

[67] Siddhartha Kumar Khaitan and James D McCalley. Design techniques and applications of cyberphysical systems: A survey. *IEEE Systems Journal*, 9(2):350–365, 2014.

[68] Martin Nečas, Petr Syrovỳ, Martin Kyncl, Tomáš Kellner, and Jiří Kyncl. Automatic quality control workplace design. 2019.

[69] Matthew NO Sadiku, Yonghui Wang, Suxia Cui, and Sarhan M Musa. Cyber-physical systems: a literature review. *European Scientific Journal*, 13(36):52–58, 2017.

[70] Ana Correia Simões, António Lucas Soares, and Ana Cristina Barros. Factors influencing the intention of managers to adopt collaborative robots (cobots) in manufacturing organizations. *Journal of Engineering and Technology Management*, 57:101574, 2020.

[71] ToyotaCZ. Tis (technical instructions sheet) for body degreasing. *TMMCZ PE*, 2021.

[72] Forschungsunion Wirtschaft-Wissenschaft. Acatech: Umsetzungsempfehlungen für das zukunftsprojekt industrie 4.0. *Abschlussbericht des Arbeitskreises Industrie*, 4, 2013.